I. INTRODUCTION

I.1 General program methodology

Structural identification – Enhancing the numerical approach of the analysis of linear elastic systems by structural identification (Fig. I.1), the computer code **MAFE** is developed for evaluating accurately the frequency and time responses of a structure to external dynamic loads.

The program contains :

- the *finite element method* and the *finite dynamic element formulation* (special circumstances) for the construction of the stiffness, mass and eventually gyroscopic matrices of the structure (spatial model);
- *eigensolution procedures* for the extraction of the natural frequencies and associated mode shapes of the undamped model (modal model);
- the *finite difference method* (mode superposition) for the time integration of the equilibrium equations (damping included), written in the modal basis of the structure (response model);
- *condensation techniques* in order to prepare the comparison of the modal parameters with the results obtained from an experimental analysis of the structure (linking).

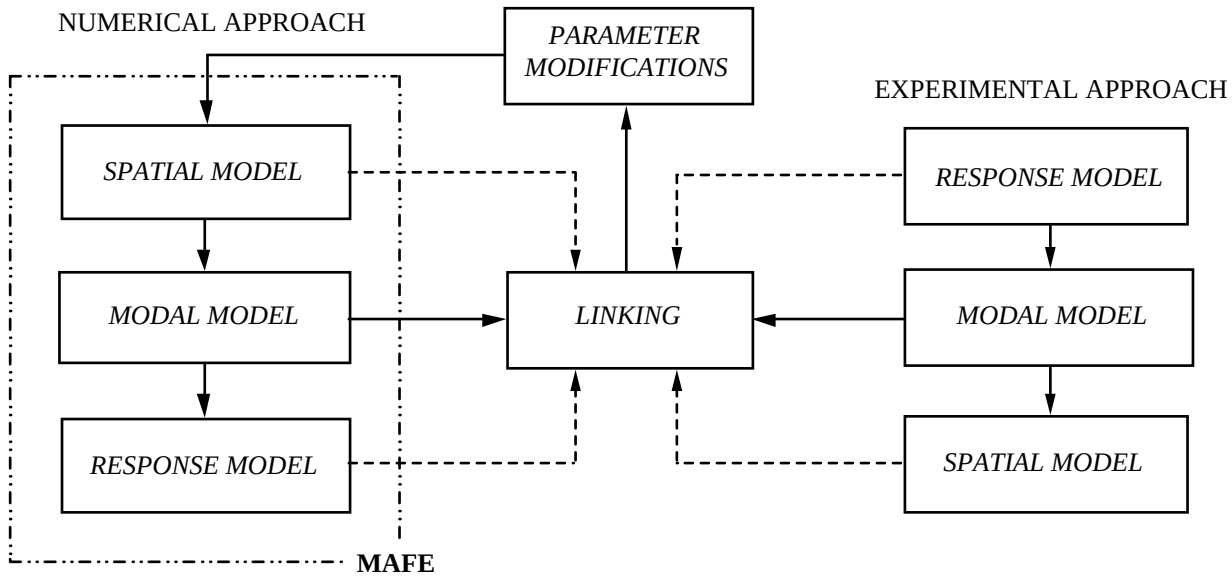


Figure I.1 Context of computer code **MAFE**.

Spatial discretisation – The dominant characteristic of the *finite element method* is to replace the actual continuous system, which has theoretically an infinite number of degrees of freedom, by an approximate multi-degree-of-freedom system. Defining the displacement field within the continuous elastic system under consideration in terms of the generalized displacements at a finite number of points (nodes) through piecewise polynomial shape functions, the equations of motion governing the linear dynamic behaviour of the structure can be written in matrix form as

$$\mathbf{M}\,\mathbf{u}^{\cdot\cdot}(t) + \mathbf{C}\,\mathbf{u}^{\cdot}(t) + \mathbf{K}\,\mathbf{u}(t) = \mathbf{f}(t) \tag{1.1}$$

$$\mathbf{u}(0) = \mathbf{u}_0\,,\ \mathbf{u}^{\cdot}(0) = \mathbf{v}_0 \tag{1.2}$$

where $\mathbf{u} = \mathbf{u}(t)$ denotes the time-dependent displacement vector, containing the n generalized coordinates of the discrete model, and a dot symbolizes differentiation with respect to the independent variable t representing time. $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{K}$ are the mass, damping and stiffness matrices of the finite element assemblage respectively and are usually large, highly banded and sparse. The vector $\mathbf{f} = \mathbf{f}(t)$ represents the vector of the external dynamic loads, and $\mathbf{u}_0$ and $\mathbf{v}_0$ denote the generalized initial displacement and velocity vectors.

It should be emphasized that generation of the damping matrix $\mathbf{C}$ presents special difficulties, due to lack of information regarding damping mechanisms and damping levels in structures. To circumvent this problem, the normal mode method of determining response can be used, the modal damping factors for all significant modes being estimated through the experimental model, as shown below.

Generalized eigenproblem – The computer code transforms the equilibrium equations (1.1) into a more effective form for direct integration by using the following transformation of the finite element nodal point displacements :

$$\mathbf{u}(t) = \Phi\,\mathbf{q}(t) \tag{1.3}$$

where Φ represents the eigenvector matrix associated to the free vibration equilibrium equations (with damping neglected)

$$\mathbf{M}\,\mathbf{u}^{\cdot\cdot}(t) + \mathbf{K}\,\mathbf{u}(t) = \mathbf{0} \tag{1.4}$$

and where the components of $\mathbf{q}$ are referred to as the generalized modal displacements.

Indeed, assuming a solution to matrix equation (1.4) of the form $\mathbf{u} = \varphi\, e^{i\omega t}$, where ω is a

real scalar and φ an n-dimensional vector, the generalized eigenproblem related to the undamped system may be written as

$$\mathbf{K}\,\Phi = \mathbf{M}\,\Phi\,\Lambda \tag{1.5}$$

where $\Lambda = \text{diag}\,(\lambda_k,\ k = 1, 2, ...n)$ is the diagonal eigenvalue matrix, storing the squares of the natural frequencies ω_k $(k = 1, 2, ...n)$ of the undamped structure, and Φ denotes the required transformation matrix, containing the mode shapes φ_k $(k = 1, 2, ...n)$ associated to the natural frequencies ω_k $(k = 1, 2, ...n)$.

In the computer program, the eigenpairs $\{(\omega_k, \varphi_k),\ k = 1, 2, ...p\}$ corresponding to the p lowest eigenvalues of problem (1.5) are extracted by efficient *eigensolution algorithms*, such as the standard or accelerated subspace iteration method and the generalized Lanczos procedure with selective orthogonalization.

The classical *subspace inverse iteration method* has been proved to be very efficient when only a subset of the lowest eigenvalues and associated eigenvectors is to be extracted from large algebraic systems with symmetric matrices. Basically, the subspace iteration process is subdivided into the following main steps (extraction of p eigenpairs from the standard pencil $(\mathbf{K}, \mathbf{M})$ of order n) :

i) Select q $(q > p >> n)$ starting iteration vectors, which span the initial subspace, and perform subspace iterations in which simultaneous inverse iteration is used on the q vectors (Ritz vectors).

ii) At the end of each inverse iteration, perform a Ritz analysis to extract optimum approximations to the required eigenvectors and to the corresponding eigenvalues (Ritz values).

iii) After convergence, verify by a Sturm sequence check that the p lowest eigenpairs have been computed.

In the accelerated version of the algorithm, the Ritz vectors are improved by overrelaxation after each inverse iteration and a shifting strategy is used in order to increase the convergence rate, especially when the number q of iteration vectors is relatively small in comparison with the number p of requested eigenpairs. Table I.1 summarizes the implementation of the standard subspace iteration algorithm.

Despite its sensitivity to roundoff, the single *Lanczos algorithm* is an effective tool for computing some outer eigenvalues and their eigenvectors. Similarly to the subspace iteration

scheme, the Lanczos method can be thought of as means of constructing an orthogonal set of

Table I.1 Standard subspace iteration method.

Data	
K, **M**	Stiffness and mass matrices of order n
Φ_j	Eigenvector matrix storing the approximate eigenvectors (Ritz vectors) $\varphi^j{}_{,k}$ (k = 1, 2, ...q) at step j
Λ_j	Diagonal matrix containing the approximate eigenvalues (Ritz values) $\lambda^j{}_{,k} = (\omega^j{}_{,k})^2$ (k = 1, 2, ...q) at step j
q	Subspace dimension
j	Iteration counter
p	Number of requested eigenvalues ($\lambda_k = \omega^2{}_{,k}$, k = 1, 2, ...p)
Initial calculations	
1.	Choose an initial shift σ if rigid body modes are present and compute $\mathbf{K}_\sigma = \mathbf{K} - \sigma\mathbf{M}$
2.	Factorize $\mathbf{K}_\sigma$ into $\mathbf{L}_\sigma \mathbf{D}_\sigma \mathbf{L}^T{}_{,\sigma}$
3.	Select q starting vectors to be the columns of Φ_0
Iteration : for j = 0, 1, ...	
1.	Solve the simultaneous equations $\mathbf{K}_\sigma \bar{}\,{}_{,}\Phi_{j+1} = \mathbf{M}\,\Phi_j$
2.	Compute the projected operators $\mathbf{K}_{j+1} = \bar{}\,{}_{,}\Phi^T{}_{,j+1}\,\mathbf{K}_\sigma \bar{}\,{}_{,}\Phi_{j+1}$ and $\mathbf{M}_{j+1} = \bar{}\,{}_{,}\Phi^T{}_{,j+1}\,\mathbf{M} \bar{}\,{}_{,}\Phi_{j+1}$
3.	Solve by the generalized Jacobi method the reduced eigenvalue problem $\mathbf{K}_{j+1}\mathbf{R}_{j+1} = \mathbf{M}_{j+1}\mathbf{R}_{j+1}\Lambda^\sigma{}_{,j+1}$ for the diagonal eigenvalue matrix $\Lambda^\sigma{}_{,j+1} = \Lambda_{j+1} - \sigma\mathbf{I}$ and the eigenvector matrix $\mathbf{R}_{j+1}$
4.	Form the new basis $\Phi_{j+1} = \bar{}\,{}_{,}\Phi_{j+1}\mathbf{R}_{j+1}$
5.	Test for convergence on the first p eigenpairs
Sturm sequence check (after last iteration l)	
1.	Compute $\mathbf{K}_\rho = \mathbf{K} - \rho\mathbf{M}$ ($\rho > \lambda^l{}_{,p}$)
2.	Factorize $\mathbf{K}_\rho$ into $\mathbf{L}_\rho \mathbf{D}_\rho \mathbf{L}^T{}_{,\rho}$ and check number of negative terms in $\mathbf{D}_\rho$

vectors, known as Lanczos vectors, for use in the Rayleigh-Ritz procedure. The main steps of the algorithm can be stated as follows :

i) Select a unit **M**-orthogonal vector (starting Lanczos vector) and generate by using the

power iteration method an increasing sequence of vectors (Krylov sequence), which span the current so-called Krylov subspace.

ii) At each step, perform a Rayleigh-Ritz analysis by projecting the kernel matrix $\mathbf{K}^{-1}\,\mathbf{M}$ into the current Krylov subspace, extract the best approximate eigenpairs* and generate an **M**-orthonormal set of vectors (the Lanczos vectors).

Note that the Rayleigh-Ritz procedure with the **M**-orthonormal basis of the Krylov subspace leads to a standard eigenproblem with a tridiagonal matrix. Moreover, it should be pointed out that the **M**-orthogonality property of the Lanczos vectors completely breaks down after a certain number of steps depending on the unit roundoff error of the computer and the distribution of the eigenvalues. In order to prevent this loss of orthogonality – which means in fact that a Ritz value has converged during the current step –, a selective orthogonalization scheme is adopted in the code. As soon as a Ritz value converges, the corresponding Ritz vector is computed and the tridiagonal matrix is deflated. Table I.2 summarizes the implementation of the Lanczos procedure.

Solution of the equilibrium equations – Using the transformation matrix associated to the change of basis (1.3) from the finite element coordinate system to the modal basis related to the eigenproblem (1.5), the equilibrium equations (1.1) and the initial conditions (1.2) become after premultiplying by Φ^{T}

$$\Phi^{\mathrm{T}}\,\mathbf{M}\,\Phi\,\ddot{\mathbf{q}}(t) \;+\; \Phi^{\mathrm{T}}\,\mathbf{C}\,\Phi\,\dot{\mathbf{q}}(t) \;+\; \Phi^{\mathrm{T}}\,\mathbf{K}\,\Phi\,\mathbf{q}(t) \;=\; \Phi^{\mathrm{T}}\,\mathbf{f}(t) \tag{1.6}$$

$$\Phi\,\mathbf{q}(0) \;=\; \mathbf{u}_0\,,\; \Phi\,\dot{\mathbf{q}}(0) \;=\; \mathbf{v}_0 \tag{1.7}$$

Taking account of the **M**-orthonormality of the eigenvector matrix, and assuming that the modal damping matrix is diagonal, equations (1.6) and (1.7) may be written as

$$\ddot{\mathbf{q}}(t) \;+\; \Xi\,\dot{\mathbf{q}}(t) \;+\; \Lambda\,\mathbf{q}(t) \;=\; \Phi^{\mathrm{T}}\,\mathbf{f}(t) \tag{1.8}$$

$$\mathbf{q}(0) \;=\; \Phi^{\mathrm{T}}\,\mathbf{M}\,\mathbf{u}_0\,,\; \dot{\mathbf{q}}(0) \;=\; \Phi^{\mathrm{T}}\,\mathbf{M}\,\mathbf{v}_0 \tag{1.9}$$

where Ξ represents the diagonal modal damping matrix.

Matrix equation (1.8) consists thus of the following n uncoupled equations (single degree-of-freedom system) :

---

* The procedure is applied on the pencil ($\mathbf{I}$, $\mathbf{K}^{-1}\,\mathbf{M}$), so that the Ritz values converge to the reciprocals of the true eigenvalues

$$\ddot{q}_k(t) + 2\xi_k \omega_k \dot{q}_k(t) + \omega^2_k q_k(t) = r_k(t) \quad (k = 1, 2, ...n) \tag{1.10}$$

where ξ_k is the kth damping ratio and $r_k(t)$ the kth component of the projection of the applied load on the modal basis.

Table I.2 Lanczos algorithm.

Data	
K, **M**	Stiffness and mass matrices of order n
Φ_j	Eigenvector matrix storing the approximate eigenvectors (Ritz vectors) $\varphi^j{}_{,k}$ (k = 1, 2, ...j) at step j
Λ_j	Diagonal matrix containing the approximate eigenvalues (Ritz values) $\lambda^j{}_{,k} = (\omega^j{}_{,k})^2$ (k = 1, 2, ...j) at step j
$\Delta^\sigma{}_{,j}$	Diagonal matrix containing the reciprocals of the shifted approximate eigenvalues (Ritz values) $\lambda^j{}_{,k} - \sigma$ (k = 1, 2, ...j) at step j (σ initial shift)
$\mathbf{Q}_j$	Matrix storing the first j+1 Lanczos vectors $\mathbf{q}_k$ (k = 0, 1, ...j) (Krylov sequence of order j+1)
$\mathbf{T}_j$	Tridiagonal matrix associated with the Lanczos recurrence
j	Krylov subspace dimension and iteration counter
p	Number of requested eigenvalues ($\lambda_k = \omega^2{}_{,k}$, k = 1, 2, ...p)

Initial calculations	
1.	Choose an initial shift σ if rigid body modes are present and compute $\mathbf{K}_\sigma = \mathbf{K} - \sigma\mathbf{M}$
2.	Factorize $\mathbf{K}_\sigma$ into $\mathbf{L}_\sigma \mathbf{D}_\sigma \mathbf{L}^T{}_\sigma$
3.	Select a unit **M**–orthonormalized starting vector $\mathbf{q}_0$ (initial Lanczos vector)

Iteration : for j = 0, 1, ...	
1.	Solve the system of equations $\mathbf{K}_\sigma \bar{}\mathbf{,q}_{j+1} = \mathbf{M}\,\mathbf{q}_j$ and compute $\hat{}\mathbf{,q}_{j+1} = \bar{}\mathbf{,q}_{j+1} - \mathbf{q}_{j-1}\,\beta_j$ ($\beta_0 = 0$)
2.	Calculate the next diagonal term of the tridiagonal matrix $\mathbf{T}_{j+1}$ associated with the current Krylov sequence $\alpha_{j+1} = \mathbf{q}^T{}_{,j}\,\mathbf{M}\,\hat{}\mathbf{,q}_{j+1}$ and compute $\tilde{}\mathbf{,q}_{j+1} = \hat{}\mathbf{,q}_{j+1} - \mathbf{q}_j\,\alpha_{j+1}$
3.	Calculate the next off-diagonal term of the tridiagonal matrix $\mathbf{T}_{j+1}$ associated with the current Krylov sequence $\beta_{j+1} = (\tilde{}\mathbf{,q}^T{}_{,j+1}\,\mathbf{M}\,\tilde{}\mathbf{,q}_{j+1})^{1/2}$
4.	Solve the reduced standard eigenproblem $\mathbf{T}_{j+1}\,\mathbf{R}_{j+1} = \mathbf{R}_{j+1}\,\Delta^\sigma{}_{,j+1}$ for the diagonal eigenvalue matrix $\Delta^\sigma{}_{,j+1} = (\Lambda_{j+1} - \sigma\mathbf{I})^{-1}$ and the eigenvector matrix $\mathbf{R}_{j+1}$
5.	If required, form the new approximations to the true eigenvectors $\Phi_{j+1} = \mathbf{Q}_j\,\mathbf{R}_{j+1}$
6.	Normalize the additional Lanczos vector $\mathbf{q}_{j+1} = \tilde{}\mathbf{,q}_{j+1} / \beta_{j+1}$
7.	Test for convergence on the first p eigenpairs and selective orthogonalization against Ritz vectors or previous Lanczos vectors

Sturm sequence check (after last iteration l)	
1.	Compute $\mathbf{K}_\rho = \mathbf{K} - \rho \mathbf{M}$ $(\rho > \lambda^l{}_{,p})$
2.	Factorize $\mathbf{K}_\rho$ into $\mathbf{L}_\rho \mathbf{D}_\rho \mathbf{L}^T{}_{,\rho}$ and check number of negative terms in $\mathbf{D}_\rho$

The *finite difference method* performs time integration of the equations of motion (1.10), step by step, by replacing time derivatives by appropriate finite difference quotients. The Newmark family retained in the program* takes the following form:

$$q,^{\bullet\bullet i+1}_{,k} + 2\xi_k \omega_k q,^{\bullet i+1}_{,k} + \omega^2_{,k} q^{i+1}_{,k} = r^{i+1}_{,k} \tag{1.11}$$

$$q^{i+1}_{,k} = q^{i}_{,k} + \tau q,^{\bullet i}_{,k} + \textbf{Erreur !}\tau^2 [(1-2\alpha)\,\textbf{Erreur !} + 2\alpha\,\textbf{Erreur !}] \tag{1.12}$$

$$q,^{\bullet i+1}_{,k} = q,^{\bullet i}_{,k} + \tau [(1-\sigma)\, q,^{\bullet\bullet i}_{,k} + \sigma q,^{\bullet\bullet i+1}_{,k}\ ] \tag{1.13}$$

where $q^{i}_{,k}$, $q,^{\bullet i}_{,k}$, and $q,^{\bullet\bullet i}_{,k}$ are the approximations at the step i of $q_k(t_i)$, $q,^{\bullet}_{k}(t_i)$, and $q,^{\bullet\bullet}_{k}(t_i)$ respectively, and τ denotes the time increment. Parameters σ and α determine the stability and accuracy characteristics of the algorithm under consideration. Note that the time increment τ must be chosen sufficiently small in order to avoid period elongation** of the response (step-to-period ratio smaller than about 0.01), even if the scheme is unconditionally stable. The Newmark family contains as specific cases many well-known and widely used methods (Table I.3).

Table I.3 Newmark family members (Ω_{crit} = critical sampling frequency for undamped case).

Method	Type	σ	α	Stability condition
Average acceleration (trapezoidal rule)	Implicit	1/2	1/4	Unconditional
Linear acceleration	Implicit	1/2	1/6	$\Omega_{crit} = 2\sqrt{,3}$
Fox-Goodwin (royal road)	Implicit	1/2	1/12	$\Omega_{crit} = \sqrt{,6}$
Central difference	Explicit	1/2	0	$\Omega_{crit} = 2$

Finally, the solution of the n equilibrium equations in the finite element coordinate system is retrieved by using equation (1.3) (mode superposition). It should be mentioned that in general only a few modes, corresponding to the lowest natural frequencies of the structure, are neces-

* The Wilson-θ method, which is a member of the collocation schemes, is also included in the computer code, but is not described in this user's manual

** In the Wilson–θ method, an amplitude decay could further be observed

sary for the assessment of a well-approximated solution. Unfortunately, no general relation between n and the number of required mode shapes to be superimposed can be given, since the latter value depends on the parameters of the structure, the nature of the excitation and the response quantity of interest.

Reduction techniques – The most popular application of modal testing is to provide a direct and objective comparison between specific measured and predicted dynamic properties of a structure, and to quantify the extent of the differences or similarities between the two sets of data. Owing to adjustments or modifications provided to one or other set of results in order to bring them closer into line with each other, the numerical model can be validated for further analysis.

Several methods of comparison, well described in the literature, have been developed to this end. Nevertheless, in order to apply these procedures, it is generally necessary to condense the finite element mass or stiffness matrix to the same coordinate set as has been used in the modal test. In the computer code, the free motion equilibrium equations (1.4) are rewritten in partitioned form as

$$\begin{bmatrix} \mathbf{M}_{mm} & \mathbf{M}_{ms} \\ \mathbf{M}_{sm} & \mathbf{M}_{ss} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{u}}_m(t) \\ \ddot{\mathbf{u}}_s(t) \end{Bmatrix} + \begin{bmatrix} \mathbf{K}_{mm} & \mathbf{K}_{ms} \\ \mathbf{K}_{sm} & \mathbf{K}_{ss} \end{bmatrix} \begin{Bmatrix} \mathbf{u}_m(t) \\ \mathbf{u}_s(t) \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \tag{1.14}$$

where the subscripts m and s refer to master and slave quantities respectively.

Assuming that the inertia effects related to the slave displacements $\mathbf{u}_s$ are negligible, these displacements can be expressed in terms of the master displacements $\mathbf{u}_m$ by expanding the equation below the partition as follows :

$$\mathbf{u}_s = -\mathbf{K}^{-1}_{ss}\ \mathbf{K}_{sm}\ \mathbf{u}_m \tag{1.15}$$

Using equations (1.14) and (1.15), the free vibration equations (1.4) can be expressed as

$$\mathbf{M}_m\ \ddot{\mathbf{u}}_m + \mathbf{K}_m\ \mathbf{u}_m = \mathbf{0} \tag{1.16}$$

with

$$\mathbf{K}_m = \mathbf{K}_{mm} - \mathbf{K}_{ms}\ \mathbf{K}^{-1}_{ss}\ \mathbf{K}_{sm} \tag{1.17}$$

$$\mathbf{M}_m = \mathbf{M}_{mm} - \mathbf{M}_{ms}\ \mathbf{K}^{-1}_{ss}\ \mathbf{K}_{sm} - \mathbf{K}_{ms}\ \mathbf{K}^{-1}_{ss}\ (\mathbf{M}_{sm} - \mathbf{M}_{ss}\ \mathbf{K}^{-1}_{ss}\ \mathbf{K}_{sm}) \tag{1.18}$$

where $\mathbf{K}_m$ and $\mathbf{M}_m$ are the so-called statically condensed stiffness and mass matrices.

Assuming a solution to matrix equation (1.16) of the form $\mathbf{u}_m = \varphi^m e^{i\omega^m t}$, where ω^m is a real scalar and φ^m an n-dimensional vector, the condensed generalized eigenproblem ($m << n$) associated to equation (1.16) may be written as

$$\mathbf{K}_m \Phi_m - \mathbf{M}_m \Phi_m \Lambda_m = \mathbf{0} \tag{1.19}$$

where Φ_m is the eigenvector matrix storing the reduced mode shapes $\varphi^m{}_{,k}$ ($k = 1, 2, ...m$) and $\Lambda_m = \text{diag}\,(\lambda^m{}_{,k}, k = 1, 2, ...m)$ denotes the diagonal eigenvalue matrix storing the squares of the natural frequencies $\omega^m{}_{,k}$ ($k = 1, 2, ...m$) .

In addition to the *static* or *Irons-Guyan reduction technique* described above, the *dynamic condensation* is also available in the computer code. Based on the standard reduction of the dynamic stiffness matrix and on the expansion in a series form of the inverse of the dynamic slave stiffness matrix, this reduction procedure leads to the following quadratic eigenproblem :

$$\mathbf{K}_m \Phi_m - \mathbf{M}_m \Phi_m \Lambda_m - \mathbf{N}_m \Phi_m \Lambda^2{}_{,m} = \mathbf{0} \tag{1.20}$$

where $\mathbf{K}_m$ and $\mathbf{M}_m$ are the static stiffness and mass matrices given by equations (1.17) and (1.18), and where $\mathbf{N}_m$, which is defined by the following expression :

$$\begin{aligned}\mathbf{N}_m = {} & \mathbf{K}_{ms}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{M}_{ss}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{M}_{ss}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{K}_{sm} - \mathbf{M}_{ms}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{M}_{ss}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{K}_{sm} \\ & - \mathbf{K}_{ms}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{M}_{ss}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{M}_{sm} + \mathbf{M}_{ms}\,\mathbf{K}^{-1}{}_{,ss}\,\mathbf{K}_{sm}\end{aligned} \tag{1.21}$$

denotes the dynamic mass matrix.

Undamped gyroscopic systems – When gyroscopic effects are included in the analysis, the equations of free motion (1.4) governing the linear dynamic behaviour of an undamped system are rewritten in matrix form as

$$\mathbf{M}\,\ddot{\mathbf{u}}(t) + \mathbf{G}\,\dot{\mathbf{u}}(t) + \mathbf{K}\,\mathbf{u}(t) = \mathbf{0} \tag{1.22}$$

where matrix **G**, representing the spinning effects which are linear functions of the angular speed of the system, is the so-called skew-symmetric gyroscopic matrix of the finite element assemblage.

If the stiffness matrix $\mathbf{K}$ is positive definite, the second-order differential equation system

(1.22) can be replaced by the $2n$ first-order equations

$$\mathbf{A}\,\mathbf{v} = \mathbf{B}\,\mathbf{v}, \tag{1.23a}$$

with

$$\mathbf{A} = \tag{1.23b}$$

where **A** and **B** are the expanded structural matrices of order $2n$, and $\mathbf{v} = \{ \mathbf{u}, \dot{\mathbf{u}} \}^T$ denodes the $2n$-dimensional state vector of the expanded differential system.

Assuming a solution to equation (1.23a) of the form $\mathbf{v} = \psi e^{\lambda t}$, where λ is a constant complex number and ψ is a $2n$-dimensional constant complex vector, the generalized eigenproblem associated with an undamped gyroscopic system may be expressed as

$$\mathbf{A}\,\Psi = \mathbf{B}\,\Psi\,\Lambda \tag{1.24}$$

where $\Lambda = \text{diag}\,(\lambda_k,\ k = 1, 2, ...2n)$ is the diagonal eigenvalue matrix and Ψ denotes the eigenvector matrix containing the expanded mode shapes ψ_k ($k = 1, 2, ...2n$). Owing to the properties of the structural matrices, the eigenvalues λ_k ($k = 1, 2 ...2n$) appear in n conjugate pairs of pure imaginary values $\pm i\omega_k$ ($k = 1, 2, ...n$), where the real scalars ω_k are the natural frequencies of the system ($i = \sqrt{-1}$). The corresponding eigenvectors ψ_k ($k = 1, 2, ..., 2n$) consist of the n pairs of complex conjugate vectors $\{ \varphi_k, i\omega_k\varphi_k \}^T$ and $\{ \varphi^*_{,k}, i\omega_k\varphi^*_{,k} \}^T$ ($k = 1, 2, ...n$), where the subvectors φ_k and $\varphi^*_{,k}$ ($k = 1, 2, ...n$) are the mode shapes related to matrix equation (1.22), the superscript $*$ symbolizing a complex conjugate quantity.

In the computer code, the eigenpairs $\{(\omega_k, \varphi_k),\ k = 1, 2, ...p\}$ corresponding to the p pairs of eigenvalues of smallest moduli of problem (1.24) are extracted by an efficient algorithm derived from the standard subspace iteration method. The numerical procedure takes full advantage of the specific nature of gyroscopic problems and reduces the eigenvalue problem to a generalized form defined by two real symmetric matrices of small order. The method preserves the banded configuration of the structural matrices **K**, **G** and **M** and avoids complex algebra right from the beginning of the iteration process. Table I.4 summarizes the implementation of this *special subspace iteration algorithm*.

Finite dynamic elements (FDE) – The finite dynamic element method has proved to be an efficient numerical procedure for the free motion analysis of continuous elastic systems and has

Table I.4 Special subspace iteration method for undamped gyroscopic systems.

Data	
K, **G**, **M**	Stiffness, gyroscopic and mass matrices of order n
Ψ_j	Eigenvector matrix storing the real and imaginary parts ${}^{r}\psi^{j}_{,k}$ and ${}^{i}\psi^{j}_{,k}$ (k = 1, 2, ...q) of the expanded approximate eigenvectors (Ritz vectors) at step j
Θ_j	Real diagonal matrix containing by pairs the squares of the moduli $\omega^{j}_{,k}$ (k = 1, 2, ...q) of the approximate eigenvalues (Ritz values) at step j
Ω_j	Real skew-symmetric block diagonal matrix storing the moduli $\omega^{j}_{,k}$ at step j
$2q$	Subspace dimension
j	Iteration counter
p	Number of requested pairs of eigenvalues ($\lambda_k = \pm i\omega_k$, k = 1, 2, ...p)
Initial calculations	
1.	Factorize **K** into $\mathbf{L}\,\mathbf{D}\,\mathbf{L}^{T}$
2.	Select $2q$ real starting iteration vectors to be the columns of Ψ_0 and partition Ψ_0 into the upper and lower quantities ${}^{a}\Psi_0$ and ${}^{b}\Psi_0$
Iteration† : for j = 0, 1, ...	
1.	Solve the simultaneous equations $\mathbf{K}\,{}^{a}\Psi^{-}_{,j+1} = -\mathbf{G}\,{}^{a}\Psi_j - \mathbf{M}\,{}^{b}\Psi_j$
2.	Compute the projected operators $\mathbf{A}_{j+1} = {}^{a}\Psi^{-T}_{,j+1}\,\mathbf{K}\,{}^{a}\Psi_j + {}^{a}\Psi^{T}_{,j}\,\mathbf{M}\,{}^{b}\Psi_j$ and $\mathbf{B}_{j+1} = {}^{a}\Psi^{-T}_{,j+1}\,\mathbf{K}\,{}^{a}\Psi^{-}_{,j+1} + {}^{a}\Psi^{T}_{,j}\,\mathbf{M}\,{}^{a}\Psi_j$
3.	Solve by the generalized Jacobi method the reduced real symmetric eigenvalue problem $\mathbf{A}^{T}_{,j+1}\,\mathbf{B}^{-1}_{,j+1}\,\mathbf{A}_{j+1}\,\mathbf{R}_{j+1} = \mathbf{B}_{j+1}\,\mathbf{R}_{j+1}\,\Theta_{j+1}$ for the diagonal eigenvalue matrix Θ_{j+1} and the eigenvector matrix $\mathbf{R}_{j+1}$
4.	Form the new basis by computing ${}^{a}\Psi_{j+1} = {}^{a}\Psi^{-}_{,j+1}\,\mathbf{R}_{j+1}$ and ${}^{b}\Psi_{j+1} = {}^{a}\Psi_{j+1}\,\Omega_{j+1}$
5.	Test for convergence on the p pairs of eigenvalues of smallest moduli
Sturm sequence check† (after last iteration l)	
1.	Compute the hermitian matrix $\mathbf{K}_\rho = \mathbf{K} + i\,\rho\,\mathbf{G} - \rho^2\,\mathbf{M}$ ($\rho > \omega^{l}_{,p}$, $i = \sqrt{-1}$)
2.	Factorize $\mathbf{K}_\rho$ into $\mathbf{L}_\rho\,\mathbf{D}_\rho\,\mathbf{L}^{*T}_{,\rho}$ and check number of negative terms in the real diagonal matrix $\mathbf{D}_\rho$

† Iteration and Sturm sequence check phases are repeated for each specified angular speed of the spinning system

shown a significant improvement in eigenpair convergence when compared with the standard finite element method. In this structural idealization technique, the displacement field within an

element is uniquely defined in terms of the generalized nodal displacements through *frequency-dependent* shape functions, which are expanded in a series of ascending powers of the natural frequencies of the structure under consideration. With this expansion, which is valid only for frequencies below a cut-off value, the resulting element stiffness and inertia matrices are expressed as functions of the vibration frequencies of the system.

When limiting the series form of the interpolation functions to the fourth order, the generalized eigenproblem associated with the free motion behaviour of structures modelled with finite dynamic elements can be written in the following quadratic matrix form :

$$\mathbf{K}\,\Phi - \mathbf{M}\,\Phi\,\Lambda - \mathbf{N}\,\Phi\,\Lambda^2 = \mathbf{0} \qquad (1.25)$$

where matrix $\mathbf{N}$, which represents the additional dynamic correction term, is the so-called quartic stiffness of half quadratic mass matrix, $\Lambda = \text{diag}\,(\lambda_k, k = 1, 2, ...2n)$ is the diagonal eigenvalue matrix and Φ denotes the eigenvector matrix storing the corresponding eigenvectors φ_k ($k = 1, 2, ...2n$).

Since matrix $\mathbf{N}$ is positive definite by nature, the quadratic eigenproblem (1.25) may be expressed in a linearized form as

$$\mathbf{A}\,\Psi = \mathbf{B}\,\Psi\,\Lambda \qquad (1.26a)$$

with

$$\mathbf{A} = \qquad (1.26b)$$

where **A** and **B** are the expanded structural matrices of order $2n$ and Ψ denotes the eigenvector matrix storing the expanded eigenvectors ψ_k (k = 1, 2, ...$2n$).

Owing to the properties of the structural matrices, the eigenvalues λ_k (k = 1, 2, ...$2n$) appear in n real positive or primary values $\lambda^+_{,k}$ (k = 1, 2, ...n) and n real negative or secondary values $\lambda^-_{,k}$ (k = 1, 2, ...n) . The associated eigenvectors consist of n pairs of real vectors $\psi^+_{,k}$ and $\psi^-_{,k}$ (k = 1, 2, ...n) respectively. The secondary eigenvalues, artificially introduced by the series expansion of the interpolation functions related to the finite dynamic element formulation, have no physical meaning and must be rejected, together with the corresponding eigenvectors. Consequently, the physical solutions of the linearized eigenproblem (1.26) are limited to the primary eigenpairs $\lambda^+_{,k} = \omega^2_{,k}$ and $\psi^+_{,k} = \{ \varphi_k, \omega^2_{,k} \varphi_k \}^T$ (k = 1, 2, ...n) , where ω_k and φ_k (k = 1, 2, ...n) are the natural angular frequencies and the corresponding mode shapes of the system.

Despite the similarity of matrix equation (1.25) to the quadratic eigenproblem derived from the dynamic reduction technique, special eigensolution algorithms must be developed, since the matrices obtained by the dynamic condensation are generally full and small. In the computer code, two effective and numerically stable procedures are available for extracting the subset of the p lowest natural frequencies and associated mode shapes of large quadratic eigenproblems. Closely related to the standard subspace forward and inverse iteration methods, these algorithms preserve the banded nature of the stiffness, mass and dynamic correction matrices, reduce the eigenpair extraction to the solution of an eigenproblem of small size and are well suited for finding multiple or closed eigenvalues. Furthermore, the procedure based on the forward iteration technique ensures a monotonic convergence from above to the required eigenpairs. The implementation of both *special subspace iteration algorithms* is summarized in Table I.5.

Table I.5 Special subspace iteration algorithm for real symmetric quadratic eigenproblems.

Data	
K, **M**, **N**	Stiffness, mass and dynamic correction matrices of order n
Ψ_j	Eigenvector matrix storing the expanded approximate eigenvectors (Ritz vectors) $\psi^j{}_{,k} = \{ \varphi^j{}_{,k} , (\omega^j{}_{,k})^2 \varphi^j{}_{,k} \}^T$ $(k = 1, 2, ...2q)$ at step j
Λ_j	Diagonal matrix containing the approximate eigenvalue (Ritz values) $\lambda^j{}_{,k} = (\omega^j{}_{,k})^2$ $(k = 1, 2, ...2q)$ at step j
Δ_j	Diagonal matrix containing the reciprocals of the approximate eigenvalues (Ritz values) $\lambda^j{}_{,k}$ $(k = 1, 2, ...2q)$ at step j
$2q$	Subspace dimension
j	Iteration counter
p	Number of requested eigenvalues ($\lambda_k = \omega^2{}_{,k}$, $k = 1, 2, ...p$)
Initial calculations	
1.	Choose an initial shift σ if rigid body modes are present and compute $\mathbf{K}_\sigma = \mathbf{K} - \sigma \mathbf{M} + \sigma^2 \mathbf{N}$ and $\mathbf{M}_\sigma = \mathbf{M} + 2\sigma \mathbf{N}$
2.	Factorize $\mathbf{K}_\sigma$ into $\mathbf{L}_\sigma \mathbf{D}_\sigma \mathbf{L}^T{}_{,\sigma}$
3.	Select $2q$ starting vectors to be the columns of Ψ_0 and partition Ψ_0 into the upper and lower quantities ${}^a\Psi_0$ and ${}^b\Psi_0$
Iteration : for $j = 0, 1, ...$	

1. Solve the simultaneous equations $\mathbf{K}_\sigma\ {}^{a}\Psi,^{-}_{j+1} = -\mathbf{M}_\sigma\ {}^{a}\Psi_j - \mathbf{N}\ {}^{b}\Psi_j$
2. Compute projected 'stiffness' operator $\mathbf{A}_{j+1} = {}^{a}\Psi,^{-T}_{,j+1}\ \mathbf{K}_\sigma\ {}^{a}\Psi_j + {}^{a}\Psi^{T}_{,j}\ \mathbf{N}\ {}^{b}\Psi_j$ (inverse iteration) or $\mathbf{A}_{j+1} = {}^{a}\Psi,^{-T}_{,j+1}\ \mathbf{M}_\sigma\ {}^{a}\Psi,^{-}_{j+1} + {}^{a}\Psi,^{-T}_{,j+1}\ \mathbf{N}\ {}^{a}\Psi_j + ({}^{a}\Psi,^{-T}_{,j+1}\ \mathbf{N}\ {}^{a}\Psi_j)^T$ (forward iteration) and projected 'mass' operator $\mathbf{B}_{j+1} = {}^{a}\Psi,^{-T}_{,j+1}\ \mathbf{K}_\sigma\ {}^{a}\Psi,^{-}_{j+1} + {}^{a}\Psi^{T}_{,j}\ \mathbf{N}\ {}^{a}\Psi_j$
3. Solve by the generalized Jacobi method the reduced eigenvalue problem $\mathbf{A}_{j+1}\,\mathbf{R}_{j+1} = \mathbf{B}_{j+1}\,\mathbf{R}_{j+1}\,\Lambda^{\sigma}_{,j+1}$ (inverse iteration) or $\mathbf{A}_{j+1}\,\mathbf{R}_{j+1} = \mathbf{B}_{j+1}\,\mathbf{R}_{j+1}\,\Delta^{\sigma}_{,j+1}$ (forward iteration) for the diagonal eigenvalue matrix $\Lambda^{\sigma}_{,j+1}\ (= \Lambda_{j+1} - \sigma\mathbf{I})$ or $\Delta^{\sigma}_{,j+1}$ $(= [\Lambda_{j+1} - \sigma\mathbf{I}]^{-1})$ and the eigenvector matrix $\mathbf{R}_{j+1}$
4. Form the new basis by computing ${}^{a}\Psi_{j+1} = {}^{a}\Psi,^{-}_{j+1}\,\mathbf{R}_{j+1}$ and ${}^{b}\Psi_{j+1} = {}^{a}\Psi_{j+1}\ \Lambda^{\sigma}_{,j+1}$ (inverse iteration) or ${}^{b}\Psi_{j+1} = {}^{a}\Psi_{j+1}\,(\Delta^{\sigma}_{,j+1})^{-1}$ (forward iteration)
5. Test for convergence on the first p primary or positive eigenvalues

Sturm sequence check (after last iteration l)

1. Compute $\mathbf{K}_\rho = \mathbf{K} - \rho\,\mathbf{M} - \rho^2\,\mathbf{N} \quad (\rho > \omega^l_{,p})$
2. Factorize $\mathbf{K}_\rho$ into $\mathbf{L}_\rho\,\mathbf{D}_\rho\,\mathbf{L}^{T}_{,\rho}$ and check number of negative terms in $\mathbf{D}_\rho$

I.2 Capabilities and main features of the code

The major features and capabilities of the computer code **MAFE** are briefly listed below, more details being available in the relevant subsections of the input instructions.

Programming – Modular program structure to enhance readability and extendability. Fortran 77 language. Dynamic storage allocation. Compacted column architecture. Bandwidth optimization by reverse Cuthill-McKee algorithm.

Node and element generators – Simple, but labor-saving, nodal and element generation capabilities. Mapping from line, surface, and volume to one- , two- and three-dimensional physical spaces. Linear, quadratic or cubic mapping. Cartesian and cylindrical coordinate systems.

Element library – Linear, quadratic and cubic isoparametric beam element. Hermitian beam element with rotatory inertia and shear deformations included. Dynamic beam element with rotatory inertia and shear deformations included. Linear, quadratic and cubic superparametric serendipian and lagrangian shell element. Linear and quadratic isoparametric serendipian and lagrangian solid element. Linear, quadratic and cubic isoparametric rotating shaft element. Linear and quadratic serendipian and lagrangian solid to shell transition element. Linear, quadratic and cubic serendipian and lagrangian solid to beam transition element. User-defined element. Full, reduced or selective integration. Isotropic linear elastic material.

Boundary conditions and loading – Free or fixed boundary conditions. Concentrated stiffnesses (elastic supports). Complex loadings synthetized by combination of multiple load vectors and load time functions. Unit, harmonic and ramp time functions. Dirac impulse. User- defined time function. Ground acceleration.

Mass and damping – Consistent mass, lumped mass by row-sum technique or special lumping technique. Concentrated masses and inertiae. Rayleigh or modal damping.

Eigenpair extraction – Standard or accelerated subspace iteration method. Single Lanczos method with selective orthogonalization. Special subspace forward and inverse iteration methods for the solution of the eigenproblem associated with the finite dynamic element

formulation. Special subspace iteration method for the solution of the quadratic eigenproblem related to undamped gyroscopic systems. Rigid body modes. Error estimates. Sturm sequence check. Calculation of lowest or intermediate frequencies.

Time integration methods – Dynamic analysis by mode superposition. Newmark family schemes. Wilson-θ method.

Condensation – Static Irons-Guyan reduction. Quadratic condensation. Manual or automatic master degree-of-freedom selection. Generalized and quadratic eigenproblem solutions by the Jacobi method and a specific subspace iteration scheme.

Postprocessing (MAFEPLOT) – Interactive mode of operation. Plot of undeformed meshes and mode shapes. Real and complex mode shape animation. Hidden lines. Time history contour plots. Frequency versus angular speed plots.

I.3 Some useful references

- Bathe K.J., Wilson E.L., 'Large Eigenvalue Problems in Dynamic Analysis', *J. Eng. Mech. Div. ASCE,* **98**, 1471-1485 (1972).
- Bathe K.J., 'Convergence of Subspace Iteration', in K.J. Bathe *et al.* (eds.), *Formulations and Numerical Algorithms in Finite Element Analysis*, MIT Press, Cambridge, 1977.
- Bathe K.J., Ramaswamy S., 'An Accelerated Subspace Iteration Method', *Comp. Methods Appl. Mech. Eng.*, **23**, 313-331 (1980).
- Bathe K.J., *Finite Element Procedures in Engineering Analysis*, Prentice-Hall Inc., Englewood Cliffs, 1982.
- Gmür T., Rodrigues J., 'The Solution of the Eigenproblem Arising in the Finite Dynamic Element Analysis by a Subspace Iteration Method', *Proc. 7th Int. Modal Analysis Conf., Las Vegas*, 1989, pp. 1394-1401.
- Gmür T., 'A Subspace Forward Iteration Method for Solving the Quadratic Eigenproblem Associated with the FDE Formulation', *Int. J. for Numer. Meth. Eng*., **29**, 935-951 (1990).
- Hughes T.J.R., *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*, Prentice-Hall International Editions, Englewood Cliffs, 1987.
- Meirovitch L., *Computational Methods in Structural Dynamics*, Sijthoff & Noordhoff, Alphen aan de Rijn, 1980.
- Parlett B.N., Scott D.S., 'The Lanczos Algorithm with Selective Orthogonalization', *Math. Comp.*, **33**, 217-238 (1979).
- Parlett B.N., *The Symmetric Eigenvalue Problem*, Prentice-Hall Inc., Englewood Cliffs, 1980.
- Przemieniecki J.S., *Theory of Matrix Structural Analysis*, McGraw-Hill Book Company, New York, 1968.
- Rodrigues J., Gmür T., 'A Subspace Iteration Method for the Eigensolution of Large Undamped Gyroscopic Systems', *Int. J. for Numer. Meth. Eng.*, **28**, 511-522 (1989).
- Rodrigues J., Gmür T., 'The Solution of Large Undamped Gyroscopic Eigensystems by a Subspace Iteration Method', *Comp. & Struct.*, **32**, 591-599 (1989).

– Weaver W., Timoshenko S.P. and Young D.H., *Vibration Problems in Engineering*, Fifth edition, Wiley and Sons, New York, 1990.

– Wilkinson J.H., *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, 1965.

II. INPUT INSTRUCTIONS

II.1 Data file structure and general remarks

This section presents the necessary input instructions for the user to execute **MAFE**. Structurally, the data file is divided into four main sections, partitioned in secondary rubrics, some of them being optional and/or repeatable. All of these datasets are labelled by an abbreviated command or keyword as follows :

1. **`prob`** **Problem description**

`titl`	Title of the problem
`mesh`	Mesh characteristics
`dyna`	Dynamic solution information
`mass`	Mass type control
`exec`	Execution command
`prnt`	Print instructions
`plot`	Plot instructions
`subf`	Substructures filenames (*optional*)
`exit`	End of problem description section

2. **`node`** **Node definition**

`coor`	Nodal point coordinates
`bcnd`	Boundary conditions (*optional*)
`cstf`	Concentrated stiffnesses (*optional*)
`midv`	Mid-surface normal vectors (*optional*)
`midn`	Normal vector connectivities (*optional*)
`cmas`	Concentrated masses (*optional*)
`subn`	Substructures mesh node information (*optional*)
`itfd`	Substructures interfaces general definition (*optional*)
`itfc`	Substructures interfaces node connectivities (*optional*)
`exit`	End of node definition section

3. **`elem`** **Element description**

`elst`	Element set information
`matl`	Material properties
`geom`	Geometrical properties (*optional*)
`sect`	Section properties (*optional*)

	`gyro`	Angular speed information (*optional*)
	`dime`	Cross-sectional dimensions (*optional*)
	`cntv`	Element connectivities (*repeatable*)
	`blck`	Element connectivities – block generator (*repeatable*)
	`exit`	End of element description section
3. **`vect`**		**Vectors definition**
	`subm`	Number of substructures retained modes
	`mode`	Definition of retained modes
	`exit`	End of element description section

4. **`algo`** **Algorithm prescription**

`cond`	Condensation (*optional*)
`eigv`	Eigensolution (*optional*)
`msup`	Mode superposition (*optional, repeatable*)
`inte`	Integration scheme
`damp`	Damping information (*optional*)
`time`	Time function (*optional, repeatable*)
`inds`	Initial displacements (*optional*)
`invl`	Initial velocities (*optional*)
`load`	Dynamic loading (*optional*)
`grnd`	Ground acceleration (*optional*)
`ppdr`	Print-out/plotting directives (*optional*)
`exit`	End of mode superposition data
`stop`	End of algorithm prescription section

A great deal of flexibility is obtained by the modular structure of the input data file. Problems can be run in sequence, provided that keyword `stop` is replaced by an `exit` command between two tasks. For each problem, several analyses (various load cases or time sequences) can be performed in a row, the control information remaining valid for all of them.

Although particular attention has been devoted in the code to the detection of input specification errors, the new user is advised, before starting the first run, to read all the instructions, paying particular attention to the appended notes, to gain an initial overview of the data required to specify an analysis task to the computer code. Furthermore, the reader should recognize that no system of units is assumed by the program. Rather, it is the user's responsibility to specify data in the consistent units of his choice. Overlooking this requirement is a common source of erroneous results.

II.2 Problem description section : keyword `prob`

```
prob
```

Information block regarding the problem description must start with the keyword `prob` *and ends with an* `exit` *command (see section II.2.8).*

II.2.1 *Title of the problem* (A80)

```
titl
title
```

Variable	Columns	Value	Note	Description
`title`	1–80	–	(1)	Title of the problem

(1) Alphanumeric characters to identify the analysis.

PROB/MESH

II.2.2 *Mesh characteristics* (I5, 4X, 6I1, 4I5)

```
mesh
nnode mcode nelst nmatl nbcnd nmidv nsubs mnods isubc
isuba

msmxt
```

Variable	Columns	Value	Note	Description
nnode	1–5	[1,99999]	(1)	Total number of nodal points
mcode	10–15	[0,1]	(2)	Master degree of freedom code (6 digits, format I1) 0 admissible global degree of freedom 1 deleted global degree of freedom
nelst	16–20	[0,100]	(3)	Number of element sets (Default : 1)
nmatl	21–25	[0,100]	(4)	Number of material property sets (Default : 1)
nbcnd	26–30	[0,99999]	(–)	Number of boundary condition records (lines) defined in section node/bcnd

`nmidv`	31–35	[0,99999]	(5)	Number of mid-surface normal vectors for shell elements and shell-type nodes of solid to shell transition elements
`nsubs`	36–40	[0,99999]		Total number of substructures
`mnods`	41–45	[0,99999]		Max number of nodes per substructure
`isubc`	46-50	[0,100]		Number of current sub-structure for charged inter-face matrices
`isuba`	51-55	[0,100]		Number of adjacent sub-structure for charged inter-face matrices
`msmxt`	56-65	[0,9999999999]		Max number of terms in subs structural matrices

(1) Auxiliary nodes of beam elements or solid to beam transition elements must be included in the count. This is also the case for additional nodes used for ease of mesh generation.

(2) The first 3 digits correpond to the global *X*-, *Y*- and *Z*-translations, and the last 3 digits to the global rotations about the *X*-, *Y*- and *Z*-axes. For shell elements and shell-type nodal points of solid to shell transition elements, the last 3 digits are related to rotational degrees of freedom referenced to the local mid-surface system. Table II.1 summarizes the master boundary condition codes for the various element types available in the computer program. Parameter `mcode` is used to conserve storage and reduce input for problems not requiring all six displacement degrees of freedom.

If a digit is equal to 1, corresponding nodal displacement does not exist in the analysis, and any reference in the data to this displacement is ignored by the computer code. The degrees of freedom of each nodal point are freed or deleted automatically by the computer code according to the master boundary condition codes given in Table II.1. Note also that for shaft elements both global transverse degrees of freedom normal to the element axis, as well as the corresponding global rotational degrees of freedom, must be free.

Table II.1 in Master degree-of-freedom codes (digits 1 to 6) for the element types available the computer code (1 = deleted, ∞ = free or deleted).

Element type	Master boundary condition code `mcode`					
	1	2	3	4	5	6
Isoparametric beam element	∞	∞	∞	∞	∞	∞
Dynamic beam element	∞	∞	∞	∞	∞	∞
Hermitian beam element	∞	∞	∞	∞	∞	∞
3-D truss element*	∞	∞	∞	∞	1	1
3-D shell element	∞	∞	∞	∞	∞	1
3-D solid element	∞	∞	∞	1	1	1
Isoparametric shaft element	∞	∞	∞	∞	∞	∞
Hermitian shaft element	∞	∞	∞	∞	∞	∞
Solid to shell transition element	∞	∞	∞	∞ or 1†	∞ or 1†	1
Solid to beam transition element	∞	∞	∞	∞ or 1†	∞ or 1†	∞ or 1†
User-defined element	∞	∞	∞	∞	∞	∞

* The truss element is equivalent to the hermitian standard or dynamic element (neglecting shear deformation effects) with zero bending and torsional moments of inertia (see section II.4)

† Code ∞ for shell- or beam-type nodes and code 1 for solid-type nodes of solid to shell and solid to beam transition elements

prob/mesh

(3) An element set is a group or series of elements of a particular type (e.g. quadratic isoparametric beams, cubic lagrangian superparametric shells, ...). All elements in a set must have the same material properties, and, in addition for beam (or truss), rotating shaft and solid to beam transition elements, the same geometrical or sectional properties. Elements defined by same properties can be broken down into more than one set for ease of input or data organization.

(4) Variable `NMATL` may not exceed the number of element sets.

(5) If shell elements or solid to shell transition elements are present in the analysis, all shell or shell-type mid-surface nodes must be associated with mid-surface vectors. Note that only one mid-surface vector can be assigned at each mid-surface node.

II.2.3 *Dynamic solution information* (415)

```
dyna
icond ieigv imsup nfreq
```

Variable	Columns	Value	Note	Description
`icond`	1–5	[0,3]	(1)	Condensation technique option 0 no reduction 1 static condensation 2 static condensation and solution of reduced standard eigenproblem 3 dynamic condensation and solution of reduced quadratic eigenproblem 4 modal synthesis, no reduction after synthesis 5 modal synthesis and residual flexibilities at interfaces 6 modal synthesis and residual flexibilities for the entire mesh 7 no global reduction, creation of interface reduced matrices for synthesis purpose 8 no global reduction, modal analysis with charged interfaces and creation of reduced matrices for synthesis purpose

`ieigv`	6–10	[0,6]	(2)	Eigensolution procedure option 0 no eigenpair extraction 1 standard subspace iteration method 2 accelerated subspace iteration method 3 generalized Lanczos method with selective orthogonalization 4 subspace inverse iteration method for quadratic FDE eigenproblems 5 subspace forward iteration method for quadratic FDE eigenproblems 6 subspace iteration method for undamped gyroscopic eigenproblems
`imsup`	11–15	[0,2]	(3)	Mode superposition option 0 no modal superposition 1 Wilson-θ method 2 Newmark scheme
`nfreq`	16–20	[0,99999]	(4)	Number of eigenpairs to be extracted (Default : 1)

PROB/DYNA

(1) When rotating shaft elements are present in the mesh, static or dynamic condensation is only performed for the system at rest. If the mesh includes finite dynamic elements (FDE), condensation techniques can not be applied.

(2) If the mesh includes finite dynamic elements (FDE), the special subspace iteration schemes adapted to the solution of quadratic eigenproblems must be used. If rotating shaft elements are present in the mesh, the eigenpairs must be extracted by means of the solution technique developed for undamped gyroscopic eigenproblems (see also sections I.1 and II.5.2). The flow-chart of each eigensolution procedure is given in section I.1.

(3) Mode superposition can only be performed after an eigenpair extraction of the generalized linear eigenproblem (`IEIGV` equal to 1, 2, or 3).

(4) The number of eigenpairs to be calculated is limited to the number of non-zero pivots in the global mass matrix. The eigensolution procedure extracts the lowest `NFREQ` eigenvalues (eventually located within a given frequency band) and the associated eigenvectors. If the mesh includes finite dynamic elements (FDE), the code computes the `NFREQ` lowest primary or positive eigenvalues and the associated eigenvectors. For undamped gyroscopic systems, the `NFREQ` pairs of pure imaginary eigenvalues of smallest moduli and the real and imaginary parts of the corresponding complex eigenvectors are calculated (see also sections I.1 and II.5.2).

II.2.4 *Mass type control* (2I5)

```
mass
imass icmas
```

Variable	Columns	Value	Note	Description
imass	1–5	[0,3]	(1)	Mass matrix option 0 no mass effects 1 diagonal mass matrix (row-sum technique) 2 diagonal mass matrix (special lumping technique) 3 consistent mass matrix
icmas	6–10	[0,1]	(2)	Flag for concentrated masses 0 no concentrated masses 1 concentrated masses are specified

(1) The lumped (diagonal) or consistent system mass matrix is constant and calculated from the element data. Variable IMASS may be equal to zero, only if concentrated masses are specified. The dynamic correction matrix (half quadratic mass matrix) of finite dynamic elements (see section I.1) is always consistent. Note that the terms corresponding to rotational degrees of freedom are equal to zero in the lumped (row-sum technique and special lumping) mass matrix of hermitian and dynamic beam elements. This is also the case for rotating shaft elements if the local *r*-coordinate is not directed towards one of the global axes. Note that the gyroscopic matrix associated with rotating shaft elements is not computed if IMASS is equal to zero. If user-defined elements are present in the mesh, variable IMASS governs the type of user-defined element mass.

(2) Additional concentrated masses can be specified at selected degrees of freedom and are input on dataset node/cmas.

prob/exec

II.2.5 *Execution command* (3I5)

```
exec
iexec ioptm istrt
```

Variable	Columns	Value	Note	Description
iexec	1–5	[0,5]	(1)	Execution command 0 data check only 1 normal execution 2–3 data check and storage of structural matrices 4–5 normal execution and storage of structural matrices 6 normal execution and storage of structural matrices and normal modes for synthesis purpose
ioptm	6–10	[0,1]	(2)	Profile optimization 0 no bandwidth reduction 1 profile optimization by reverse Cuthill-McKee algorithm
istrt	11–15	[0,nnode]	(3)	Starting pseudo-peripherical node for profile optimiza-tion

(1) For data check mode (option 0), the computer code reads and prints the input data without actually solving for the solution response. In addition, the program checks the various input parameters and storage requirements for the analysis. This solution mode should be chosen prior to any execution. With the normal mode of program execution (option 1), the code performs data and storage checks and then solves for the solution response. If one of the options corresponding to the data check mode with computation and storage of the structural matrices is chosen (option 2 or 3), the code verifies the input data consistency, checks the storage requirements, calculates the stiffness, mass and eventually gyroscopic or dynamic correction matrices of the finite element assemblage and stores the computed values (upper triangular matrices in skyline mode), as well as the addresses of the

diagonal terms in the structural matrices, on the special print-out unit as given in section `prob/PRNT`. If the option 4 or 5 is selected, the code performs data and storage checks, computes the structural information mentioned above, saves the computed values (upper triangular matrices in skyline mode) on the special print-out unit as given in section `prob/PRNT` and, finally, solves for the solution response.

With options 2 and 4, the structural information is stored formatted. When using option 3, 5 or 6, the values are written unformatted on the selected tape in the following sequence: the total number of equations followed by the number of elements below the skylines of the stiffness, the mass and the gyroscopic or dynamic correction matrices, the adresses of the diagonal terms in the stiffness matrix (the last address corresponds to the total number of elements below the skyline of the stiffness matrix plus one), the components of the stiffness matrix, the components of the mass matrix, the adresses of the diagonal terms in the gyroscopic or dynamic correction matrix (same structure as for the stiffness matrix), and finally the components of the gyroscopic or dynamic correction matrix. Note that if the unformatted mode is chosen, no other information can be written on the special print-out unit; see dataset `prob/prnt` and sections II.5.1 (note 1) and II.5.2 (note 9).

(2) By this option, the profile (number of elements below skyline) of the stiffness, mass and possibly gyroscopic or dynamic correction matrices can be drastically reduced, which results in significant savings in computation time and core storage. If the optimization process fails, original node numbering is used for further analysis.

(3) Computation time for profile optimization can be decreased by choosing an appropriate starting pseudo-peripherical node (initial node of the so-called level structure). By default, this node is defined by the computer code. This information is not read if bandwidth reduction is avoided (`IOPTM` equal to zero).

PROB/PRNT

II.2.6 *Print instructions* (6I5)

```
prnt
iprnt itape ipdis ipvel ipacc ipmod
```

Variable	Columns	Value	Note	Description
iprnt	1–5	[0,3]	(1)	Data print flag 0 detailed print of input, execution and output data 1 selected print of input data and detailed print of execution and output data 2 no print of input data, detailed print of execution and output data 3 only detailed print of output data 4 detailed print of output data and synthesis
itape	6–10	[0,99]	(2)	Special print-out unit number (Default : see note)
ipdis	11–15	[0,1]	(3)	Displacement print-out code 0 no displacement print-out 1 displacements and mode shapes are printed 2 displacements and mode shapes and normal vectors are printed
ipvel	16–20	[0,1]	(3)	Velocity print-out code 0 no velocity print-out 1 velocities are printed
ipacc	21–25	[0,1]	(3)	Acceleration print-out code 0 no acceleration print-out 1 accelerations are printed

`ipmod`	26–30	[0,99999]	(4)	Mode-shape print-out code 0 no mode-shape print-out 1 mode shapes are printed 2 mode shapes statistics are printed

(1) If a large mesh is used in the analysis, it may not be necessary to print all the generated nodal point data and equation numbers. The listing of the above data can be skipped using the data print flag.

(2) Specific information, such as the components of the structural matrices (see section II.2.5, note 1), the reduced master stiffness and mass matrices obtained by the static reduction technique (see section II.5.1, note 1) or the mode shapes for a restart procedure (see section II.5.2, note 9) can be saved by indicating a special unit or tape number ; note that the file units 50 (input), 60 (output), 70 (orthogonalization of the Ritz vectors in the eigensolution schemes) and 71 (starting iteration vectors for the eigensolution schemes) are reserved by the code. By default (special print-out unit number is not specified), the storage instructions are ignored by the code without a diagnostic message. If the unformatted mode is chosen for the storage of the components of the structural matrices (see section II.2.5, note 1), no other information can be written on the special print-out unit.

(3) For the print-out of displacements, velocities and/or accelerations at specific nodal points and time steps, see the print-out/plotting directives given on dataset `algo/msup/ppdr`.
If ipdis is set to 2 and normal vectors are defined in the mesh, then tese are printed in the .mod file before the eigendatas :

```
nnode
((vimdn(i,j),i=1,3),j=1,nnode)
```

(4) The converged eigenvectors (mode shapes) are only printed if the mode-shape print-out code `IPMOD` is set to 1. The number of printed mode shapes is then equal to the number of converged eigenvalues.
The statistics on the coefficients of the mode shapes are printed in the .coe file. The following informations are available for each mode :

```
imode, frequency, mean of squares, standard deviation, # predominant coefficients
coefficients
```

prob/plot

II.2.7 ***Plot instructions*** (6I5)

```
plot
jplot jtape jpdis jpvel jpacc jpmod jplhp
```

Variable	Columns	Value	Note	Description
jplot	1–5	[0,1]	(1)	Plot creation flag 0 no plot file creation 1 plot file created for further analysis with the postprocessor **MAFEPLOT**)
jtape	6–10	[0,99]	(2)	Plot file unit (Default : 80)
jpdis	11–15	[0,1]	(3)	Displacement save code 0 displacements not saved 1 displacements saved on plot file
jpvel	16–20	[0,1]	(3)	Velocity save code 0 velocities not saved 1 velocities saved on plot file
jpacc	21–25	[0,1]	(3)	Acceleration save code 0 accelerations not saved 1 accelerations saved on plot file
jpmod	26–30	[0,1]	(4)	Mode-shape save code 0 mode shapes not saved 1 mode shapes saved on plot file

`jplhp`	31-35	[-2,0,1]	(5)	Special mode-shapes code 0 no special mode-shapes 1 mesh and mode-shapes are stored in xmesh fmt -2 rigid body and constraint modes are generated

(1) The creation option is used to store data for the postprocessor **MAFEPLOT**. See the specific user's manual for more information about the plotting capabilities.

(2) The plot file unit number `jtape` may not be equal to the special print-out unit number `itape`as given on dataset `prob/prnt`. The values reserved by the code (file units 50, 60, 70 and 71) must also be avoided. See section II.2.6 (note 2) for a short description of the reserved file units.

(3) Displacements, velocities and/or accelerations are saved at the specific nodal points and time steps defined through the print-out/plotting directives given on dataset `algo/msup/ppdr`.

(4) The converged eigenvectors (mode shapes) are only printed if the mode-shape save code `jpmod` is set to 1. The number of saved mode shapes is then equal to the number of converged eigenvalues.

(5) To generate constraint modes for a mesh with fixed interfaces definition (mcode=3), the .mod file of the corresponding free interfaces mesh is needed. Its extension has to be .hpl.

node/subf

II.3.6 Substructures filenames (*optional* a20)

Skip this section if `icond` *is not equal to 4 on dataset* `prob/dyna`.

subf
subfn

Variable	Columns	Value	Note	Description
subfn	1–20		(1)	Substructure filename (root)

(1)

prob/exit

II.2.8 *End of problem description section*

```
exit
```

II.3 Node definition section : keyword `node`

```
node
```

Information block regarding the node definition must start with the keyword `node` *and ends with an* `exit` *command (see section II.3.7).*

II.3.1 *Nodal point coordinates* (2I5, 3F10.0, 2I5)

```
coor
inode ictyp xcoor ycoor zcoor igene nstep
```

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,nnode]	(1)	Node number
`ictyp`	6–10	[0,2]	(2)	Coordinate system type 0 cartesian system 1 cylindrical system 2 cylindrical system definition
`xcoor`	11–20	]–∞,∞[	(3)	*X*- or *z*-coordinate for carte-sian or cylindrical system
`ycoor`	21–30	]–∞,∞[	(3)	*Y*- or *r*-coordinate
`zcoor`	31–40	]–∞,∞[	(3)	*Z*- or θ-coordinate (θ in de-grees)
`igene`	41–45	[0,7]	(4)	Generation option 0 no generation 1 line or circle generation 2 first order generation over a surface

(cont.)

node/coor

(cont.)

Variable	Columns	Value	Note	Description
				3 second order generation over a surface 4 first order generation within a volume 5 second order generation within a volume 6 second order generation along a line 7 third order generation along a line
nstep	46–50	[–9999,99999]	(5)	Node number increment for coordinate generation (Default : 1)

(1) Nodal coordinates must be defined for all nodes, directly or by means of a generation option. The sets need not be input in node order sequence. Last node that is input must be nnode, and no more sets are read after the set for inode equal to nnode has been input. If in the input sequence the information for a node is repeated, the last read or generated information pertaining to that node takes precedence.

(2) The cylindrical coordinate system (z, r, θ) is provided for user convenience, only for locating nodes in the global cartesian system. No other references to the cylindrical system are implied. Two records (lines) – which can not be automatically generated – are necessary to characterize a cylindrical coordinate system by option 2: the origin $O(X, Y, Z)$ (first record) and orientation $\mathbf{z}(X, Y, Z)$ (second record) of the cylindrical system z-axis, expressed in the global cartesian coordinate system; note that an auxiliary or effective node must be chosen as origin and that θ is zero in the local x-z plane (Fig. II.1). The current cylindrical coordinate system is then valid until a new system is specified. If no cylindrical system definition is given and ictyp is set to 1, the z-axis coincides with the global X-axis, and θ is zero in the global (X-Y) plane.

(3) Only right-handed coordinate systems are allowed to describe the nodal point coordinates. Note that all nodes of rotating shaft elements must be aligned with the rotation axis of the gyroscopic system and that this axis must be parallel to one of the global axes if concentrated masses are specified (see section II.3.6).

(4) For line (or circle) and first order block generations over a surface and within a volume (igene equal to 1, 2 or 4), the number of sets to be defined must be equal to 2, 4 and 8 respectively, since a linear, bilinear or trilinear master isoparametric element is used as basis of the generation scheme (Tables II.2a, II.2b and II.2c).

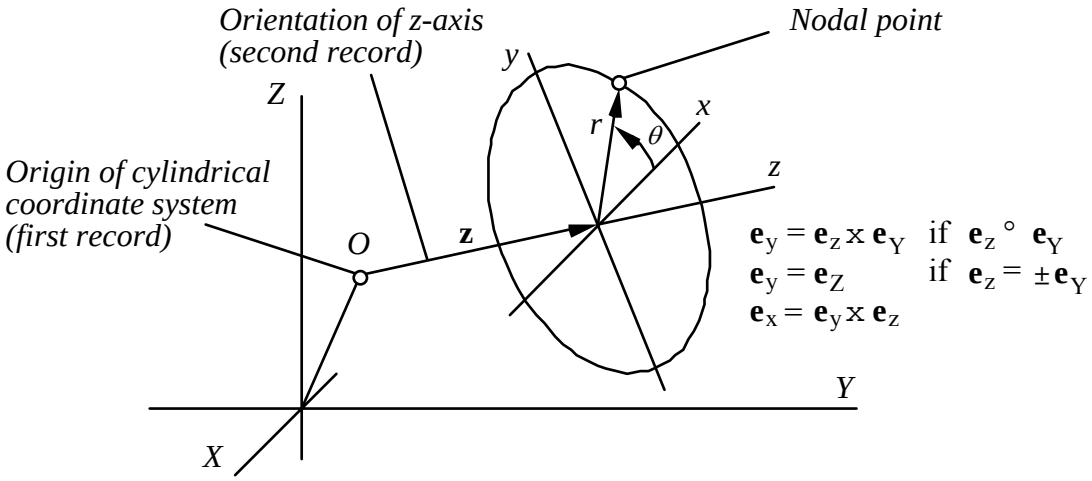


Figure II.1 Cylindrical coordinate system and nodal point location.

For second order generations over a surface and within a volume (`igene` equal to 3 or 5), the number of sets must be equal to 8 or 20 respectively, since a quadratic serendipian master 2-D or 3-D element is used as generation basis (Tables II.2d and II.2e). The nodal coordinates are generated along equal increments in the natural coordinates. With the quadratic master element, a grading and curving of the mesh can be obtained, whenever the midside generation points are purposedly off-centered. Nevertheless, midside points must coincide with nodal points and lie between 1/4 and 3/4 of the chord joining the side extremities with moderate offset.

For second and third order generations along a line (`igene` equal to 6 or 7), the number of sets must be equal to 3 or 4 respectively, since a quadratic or cubic 1-D element is used as generation basis (Table II.2f). The nodal coordinates of the missing nodes are generated along equal increments in the natural coordinate of the master element. Graded nodal spacing can be achieved by placing the internal generation points off-center. However, these points must coincide with nodal points and must ensure an invertible mapping.

Note that, from the first to the penultimate node of a generation set, parameter `igene` may not be altered, and that the last node may be used as first node of a new generation set. When the generation option is used, all the nodes in the actual generation set must be referred to the same coordinate system (cartesian or cylindrical). It should also be pointed out that the second and third order generation sche-mes may not be used with the cylindrical coordinate system.

(5) Parameter `nstep` must be specified only for the first node (first, second and third order line generation and circle generation) or the first two (generation over a surface) or three (generation within a volume) nodes of a generation set and defines the node number increment in the first (local numbering 1-2), second (2-3) and third (1-5) generation directions (see Table II.2 for the local numbering of the master nodes). Note that the nodes may be generated increasingly (`nstep`

positive) or decreasingly (`nstep` negative).

`node/coor`

Table II.2a Straight line or circle generation.

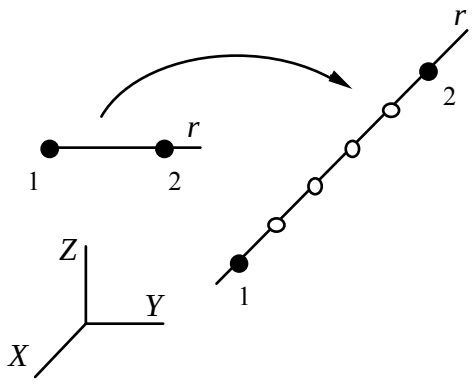
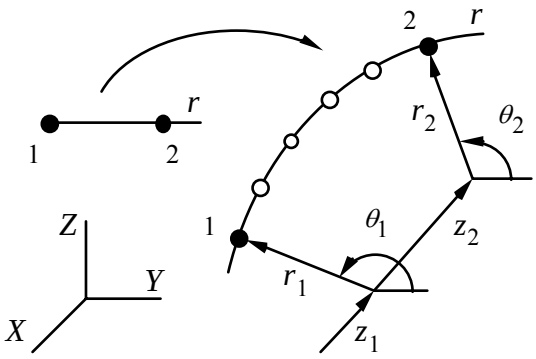
Cartesian system	Cylindrical system
	
Mapping from linear master element to straight line	Mapping from linear master element to circular line

Table II.2b First order generation over a surface.

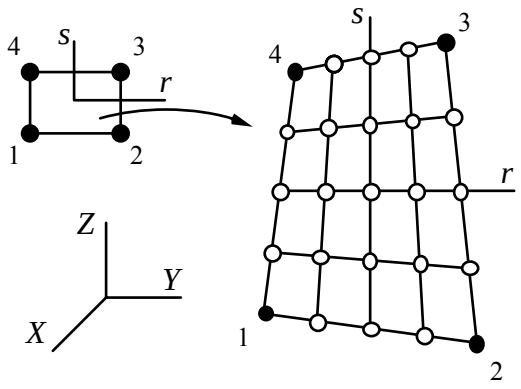
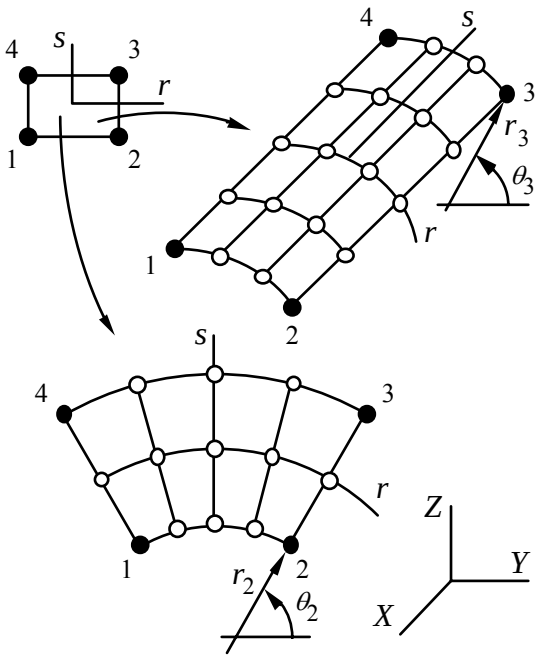
Cartesian system	Cylindrical system
	
Mapping from bilinear master element to plane or curved 3-D quadrilateral with straight edges	Mapping from bilinear master element to plane or curved 3-D quadrilateral with 2 straight and 2 curved edges

Table II.2c First order generation within a volume.

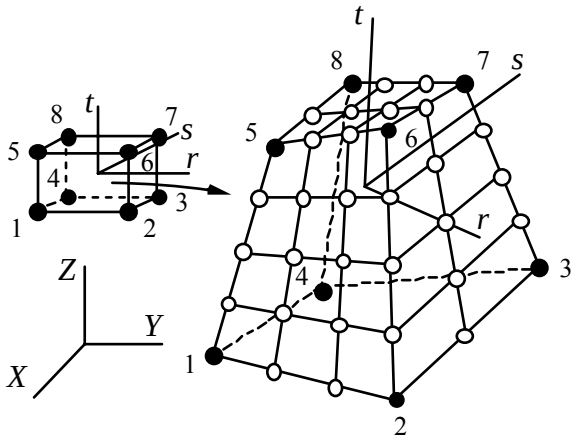
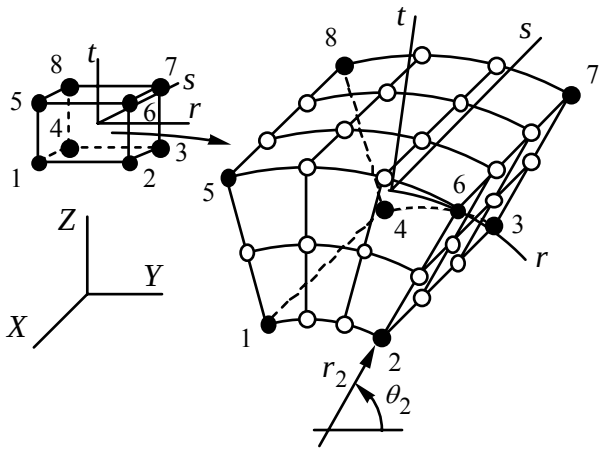
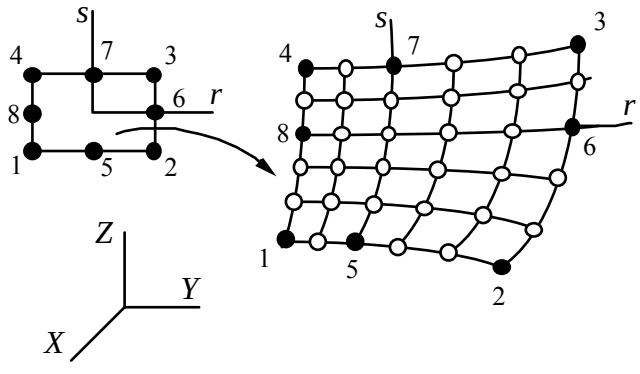
Cartesian system	Cylindrical system
	
Mapping from trilinear master element to brick shaped volume with straight edges	Mapping from trilinear master element to brick shaped volume with 8 straight and 4 curved edges

Table II.2d Second order generation over a surface (cartesian system*).

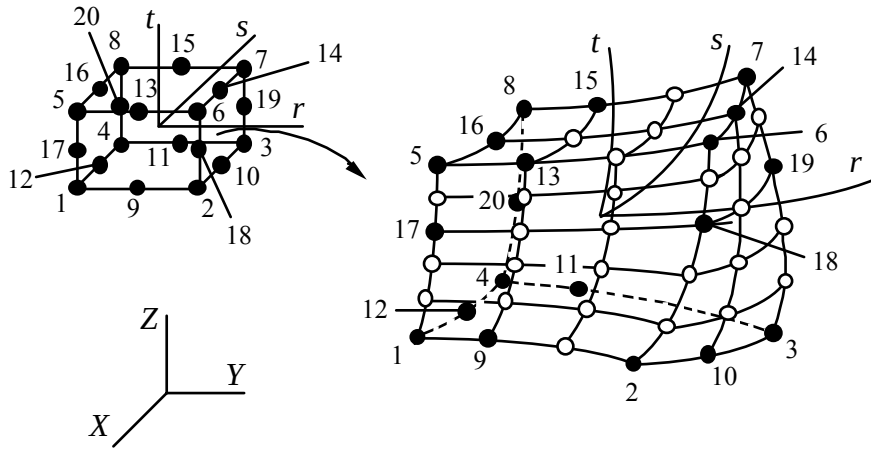


Mapping from quadratic serendipian master element to plane or curved 3-D quadrilateral with straight or curved edges

* Cylindrical system not allowed

node/coor

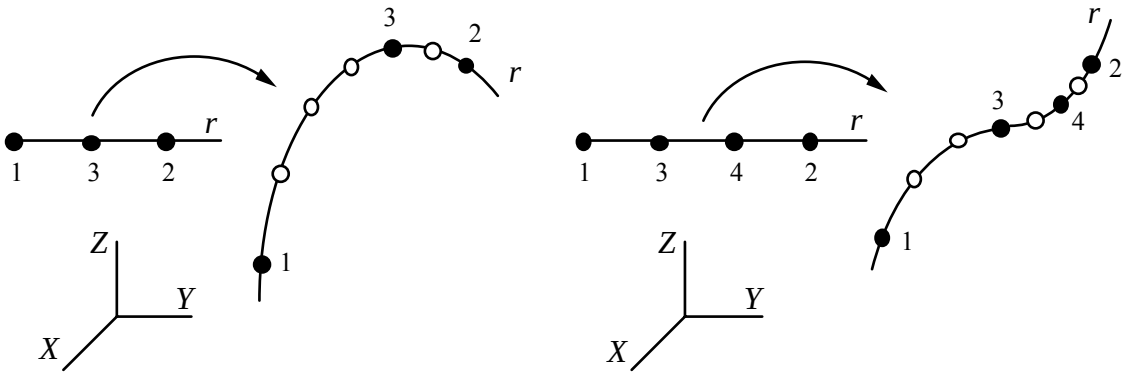
Table II.2e Second order generation within a volume (cartesian system*).



Mapping from quadratic serendipian master element
to brick shaped volume with straight or curved edges

* Cylindrical system not allowed

Table II.2f Second and third order generations along a line (cartesian system*).



Mapping from quadratic or cubic master element to curved line

* Cylindrical system not allowed

node/bcnd

II.3.2 ***Boundary conditions*** (*optional* / I5, 4X, 6I1, 2I5)

Skip this section if `nbcnd` *is equal to zero on dataset* `prob/mesh`.

```
bcnd
inode icode igene nstep
```

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,nnode]	(1)	Node number
`icode`	10–15	[0,2]	(2)	Nodal boundary condition code (6 digits, format I1)
				0 free displacement
				1 fixed displacement
				2 elastic displacement
				3 fixed interface node
`igene`	16–20	[0,2]	(3)	Generation option
				0 no generation
				1–2 generation
`nstep`	21–25	[–9999,99999]	(3)	Node number increment for boundary condition generation (Default : 1)

(1) The number of sets (lines or records) to be input is equal to the value of parameter `Nbcnd` defined on dataset `prob/mesh`. Only nodes with one or more non-zero boundary conditions associated with displacements which are not deleted by the master degree-of-freedom code `mcode` on dataset `prob/mesh` nor automatically fixed by the code (see section II.2.2, note 2) must be specified; however, master code `mcode` may not be overwritten. If in the input sequence the information for a node is repeated, the last read or generated information pertaining to that node takes precedence. The sets need not be input in node order sequence. Note that the degrees of freedom of the nodes which do not pertain to the finite element mesh, i. e. nodes which have been introduced for ease of mesh generation or auxiliary nodal points of beam elements and solid to beam transition elements, must be deleted.

(2) The first 3 digits correspond to the global *X*-, *Y*- and *Z*-translations, and the last 3 digits to the global rotations about the *X*-, *Y*- and *Z*-axes. For shell elements and

shell-type nodes of solid to shell transition elements, the latter are related to rotational degrees of freedom referenced to the local mid-surface system.

node/bcnd

If option 2 is chosen, the node is assumed to lie on an elastic support in the direction of the selected degree of freedom. The stiffness of this elastic support is then defined on dataset `node/cstf`.

(3) The boundary conditions of the generated nodes are set equal to those of the first or second node of the generation set, according as parameter `igene` is equal to 1 or 2 respectively. Obviously, variable `nstep` must be compatible with the generation scheme. Note that the nodes may be generated increasingly (`nstep` positive) or decreasingly (`nstep` negative) and that the second node of the generation set may be used as first node of a new generation scheme.

node/cstf

II.3.3 *Concentrated stiffnesses* (*optional* / 2I5, F10.0, 2I5)

Skip this section if no boundary condition is equal to 2 *on dataset* `node/bcnd`.

```
cstf
inode idirc stiff igene nstep
```

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,`nnode`]	(1)	Node number
`idirc`	6–10	[0,6]	(2)	Global degree of freedom number of the elastic support
`stiff`	11–20	[0.,∞]	(3)	Stiffness of elastic support in the direction `idirc`
`igene`	21–25	[0,2]	(4)	Generation option 0 no generation 1 line generation 2 block generation
`nstep`	26–30	[–9999,99999]	(5)	Node number increment for boundary condition generation (Default : 1)

(1) The number of concentrated stiffnesses to be defined (directly or by means of the generation option) must be equal to the total number of elastic displacements (explicitly defined and generated) specified on dataset `node/bcnd`; otherwise, a reading error will occur. The sets need not be input in node order sequence; however, a stiffness information may not be overwritten by a further set.

(2) For shell elements and shell-type nodes of solid to shell transition elements, the rotational degrees of freedom 4 to 6 are referenced to the local mid-surface system.

(3) If variable `stiff` is set to zero or a high amplitude, the associated elastic displacement becomes free or fixed respectively. Note that the stiffness of the elastic support must be positive.

node/cstf

(4) For line or block generation, the number of sets to be defined must be equal to 2 or 4 respectively. The generation schemes are identical to the straight line generation option and the first order block generation option described in section `node/ coor`. The stiffness associated to missing nodes is then determined by linearly interpolating the data input for these two or four sets. Note that parameter `idirc` may not be altered in the generation set. Furthermore, from the first to the penultimate node of a generation set, variable `igene` may not be changed, but the last node may be used as first node of a new generation scheme.

(5) Parameter `nstep` must be defined only for the first (line generator) or first two (block generator) nodes of a block generation set and be compatible with the generation scheme. Nodes may be generated increasingly (`nstep` positive) or decreasingly (`nstep` negative).

node/midv

II.3.4 ***Mid-surface normal vectors*** (*optional* / 2I5, 4F10.0, 2I5)

Skip this section if `nmidv` *is equal to zero on dataset* `prob/mesh`.

```
midv
imidv imtyp xmidv ymidv zmidv thick igene mstep
```

Variable	Columns	Value	Note	Description
`imidv`	1–15	[1,`nmidv`]	(1)	Mid-surface normal vector number
`imtyp`	6–10	[0,1]	(2)	Mid-surface normal vector definition code 0 projections onto the global axes are used 1 Euler angles are used
`xmidv`	11–20	]–∞,∞[	(2)	Projection of the vector onto the global *X*-axis or first Euler angle φ about *Z*-axis
`ymidv`	21–30	]–∞,∞[	(2)	Projection of the vector onto the global *Y*-axis or second Euler angle θ about *X*'-axis
`zmidv`	31–40	]–∞,∞[	(2)	Projection of the vector onto the global *Z*-axis
`thick`	41–50	]0.,∞[	(3)	Shell thickness associated with current mid-surface normal vector
`igene`	51–55	[0,1]	(4)	Generation option 0 no generation 1 generation
`mstep`	56–60	[–9999,99999]	(4)	Mid-surface normal vector number increment

node/midv

(1) Mid-surface normal vectors are not necessarily normal to the shell or solid to shell transition mid-surface and should hence be called director vectors. Note that the shell and solid to shell transition element formulation is based on the assumption that the normal stresses in the direction of these vectors are equal to zero. Furthermore, the director vectors associated with a shell element or the shell-type nodes of a solid to shell transition element also govern the input sequence of the nodal point numbers in the definition of the element; namely, all these vectors and the element local *t*-coordinate axis must be directed towards the same surface of the element (Fig. II.2).

The number of mid-surface normal vectors to be defined (directly or by means of the generation option) must be equal to the value of parameter `nmidv` on dataset `prob/mesh`. The sets need not be input in mid-surface vector order sequence. Input data must end with the *`nmidv`*th vector, which may not be generated. If in the vector input sequence the information for a vector is repeated, the last read or generated information is retained.

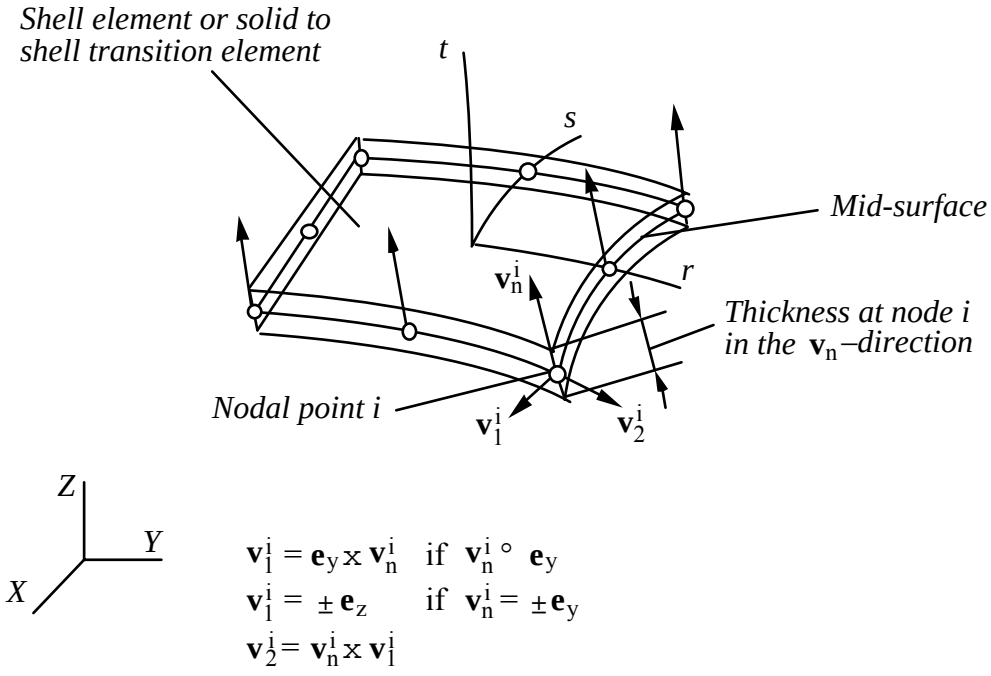


Figure II.2 Mid-surface normal vector $\mathbf{v}_{n,}^{i}$ at node *i* and shell thickness in the $\mathbf{v}_{n,}^{i}$ –direction.

(2) Mid-surface director vectors may be defined either by projections onto the global axes or by Euler angles. Note that the vectors are normalized by the code. With the Euler angle option, `xmidv` is the rotation φ about the *Z*-axis (Fig. II.3), which results into the (*X*', *Y*', *Z*') system. `ymidv` is then the rotation θ about the *X*'-axis and results in the (*X*", *Y*", *Z*") system, the axis *Z*" coinciding with the direction of the director vector $\mathbf{v}_{n,}^{i}$ at node *i*. For the Euler angle option, `zmidv` is not used.

(3) The shell or solid to shell transition thickness at a given node is measured in the direction of the normal or director vector associated to that node (Fig. II.2).

(4) The generation option can be used to generate normal vectors and/or shell and solid to shell transition thicknesses. However, normal vectors can only be generated if they are defined by Euler angles; the missing vectors are then constructed by linearly interpolating the angles φ and θ from the data input for the two mid-surface sets used for the generation scheme. Note that the vectors may be generated increasingly (`mstep` positive) or decreasingly (`mstep` negative) and that the second mid-surface normal vector of the generation set may be used as first one of a new generation scheme.

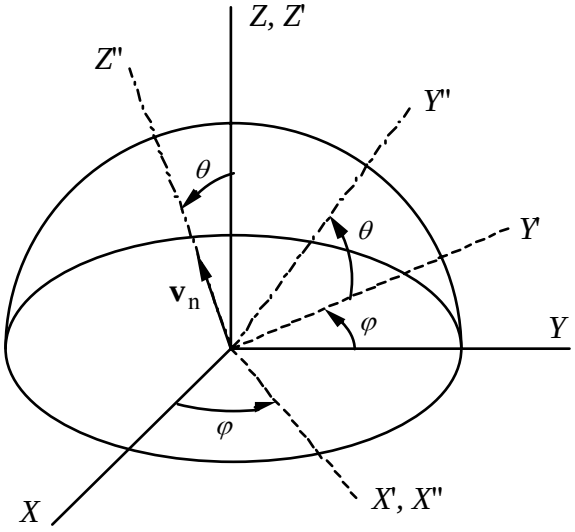


Figure II.3 Definition of director (normal) vectors by Euler angles.

node/midn

II.3.5 *Normal vector connectivities* (*optional* / 5I5)

Skip this section if `nmidv` *is equal to zero on dataset* `prob/mesh`.

```
midn
inode imidv igene nstep mstep
```

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,`nnode`]	(1)	Node number
`imidv`	6–10	[1,`nmidv`]	(1)	Mid-surface normal vector number
`igene`	11–15	[0,2]	(2)	Generation option 0 no generation 1 line generation 2 block generation
`nstep`	16–20	[–9999,99999]	(3)	Node number increment for normal vector connectivity generation (Default : 1)
`mstep`	21–25	[–9999,99999]	(3)	Mid-surface normal vector number increment

(1) Mid-surface director or normal vectors (defined on dataset `node/midv`) are only assigned to shell mid-surface nodes or shell-type nodes of solid to shell transition elements. Input data is read up to next keyword. The sets need not be input in node order sequence.

(2) For line or block generation, the number of sets to be defined must be equal to 2 or 4 respectively. The generation schemes are identical to the straight line generation option and the first order block generation option described in section `node/ coor`. Note that, from the first to the penultimate node of a generation set, variable `igene` may not be altered, and that the last node may be used as first node of a new generation set.

(3) Parameters `nstep` and `mstep` must be defined only for the first (line generator) or first two (block generator) nodes of a block generation set and be compatible with the generation scheme. Nodes and mid-surface vectors may be generated increasingly or decreasingly.

II.3.6 ***Concentrated masses*** (*optional* / I5, 6F10.0, 2I5)

Skip this section if `Icmas` *is equal to zero on dataset* `prob/MASS`.

```
cmas
inode xmass ymass zmass xinrt yinrt zinrt igene nstep
```

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,`nnode`]	(1)	Node number
`xmass`	6–15	[0.,∞[	(2)	*X*-translation mass
`ymass`	16–25	[0.,∞[	(2)	*Y*-translation mass
`zmass`	26–35	[0.,∞[	(2)	*Z*-translation mass
`xinrt`	36–45	[0.,∞[	(2)	*X*-rotation inertia
`yinrt`	46–55	[0.,∞[	(2)	*Y*-rotation inertia
`zinrt`	56–65	[0.,∞[	(2)	*Z*-rotation inertia
`igene`	66–70	[0,1]	(3)	Generation option 0 no generation 1 generation
`nstep`	71–75	[–9999,99999]	(3)	Node number increment for concentrated mass generation

(1) Nodes that are not defined in this section have zero generalized concentrated masses added to the structural lumped or consistent mass and possibly gyroscopic matrices due to the distributed element mass and gyroscopic effects. The sets need not be selected in node order sequence. The last node that is input must be `nnode` and no more sets are read after the set for `inode` equal to `nnode` has been input.

(2) If mass components are input for deleted degrees of freedom, these masses are ignored by the program without a diagnostic message, and if a node is repeated in the input, the masses specified accumulate at the nodal degrees of freedom. For shell nodes and shell-type nodal points of solid to shell transition elements, `xinrt` and `yinrt` are referenced to the local mid-surface system. For gyroscopic systems (rotating shaft elements) with generalized concentrated masses, the rotation axis must be aligned with one of the global axes; moreover, both transverse bending inertiae must be identical in order to properly assemble the gyroscopic matrix. Consistency of translation mass information is not controlled

by the code.

node/cmas

(3) With the generation option, the magnitude of the masses associated to missing nodes is determined by linearly interpolating the data input for the two sets used for the generation scheme. Note that the nodes may be generated increasingly or decreasingly and that the second node of the generation set may be used as first node of a new generation scheme.

node/subn

II.3.6 Substructures mesh node information (*optional* / 3I5)

Skip this section if `icond` *is not equal to 4 on dataset* `prob/dyna`.

```
subn
isubc nnods nsitf
```

Variable	Columns	Value	Note	Description
`isubc`	1–5	[1,`nsubs`]	(1)	Substructure number
`nnods`	6–10	[1,nmidv]	(1)	Number of nodes
`nsitf`	11–15	[1,ntitf[	(1)	Number of interfaces

(1)

node/itfd

II.3.6 Substructures interfaces general definition (*optional* / 3I5)

Skip this section if `icond` *is not equal to 4 on dataset* `prob/dyna`.

```
itfd
isubc isuba ninod
```

Variable	Columns	Value	Note	Description
`isubc`	1–5	[1,`nsubs`]	(1)	Current substructure number
`isuba`	6–10	[1,`nsubs`]	(1)	Adjacent substructure number
`ninod`	11–15	[`1`,`nmidv`[	(1)	Number of interface nodes

(1)

`node/itfc`

II.3.6 Substructures interfaces node connectivities (*optional* / 3I5)

Skip this section if `icond` *is not equal to 4 on dataset* `prob/dyna`.

`itfc`
`inode igene nstep`

Variable	Columns	Value	Note	Description
`inode`	1–5	[1,nmidv]	(1)	Node number
`igene`	6–10	[0,1]	(4)	Generation option 0 no generation 1 line generation
`nstep`	11–15	[–9999,99999]	(5)	Node number increment for interface node generation (Default : 1)

(1)

node/exit

II.3.7 *End of node definition section*

```
exit
```

Variable	Columns	Value	Note	Description

II.4 Element description section : keyword `elem` (only for fem processing)

```
elem
```

Information block regarding the element description must start with the keyword `elem` *and ends with an* `exit` *command (see section II.4.9).*

II.4.1 *Element set information* (3I5, 2X, 3I1, I5)

```
ELST
IELST IETYP Nelem INTEG IMATL
```

Variable	Columns	Value	Note	Description
`IELST`	1–5	[1,`NELST`]	(1)	Element set number
`IETYP`	6–10	[–]	(2)	Element type
`Nelem`	11–15	[1,99999]	(–)	Total number of elements in current set
`INTEG`	16–20	[1,3]	(3)	Integration code (3 digits, format I1; default : 3) 1 Reduced integration 2 Selective integration 3 Full integration
`IMATL`	21–25	[0,`NMATL`]	(4)	Material property set number (Default : 1)

(1) Element sets must be input in ascending order and may not be generated.

(2) All elements in a set must be of same type. Allowed values for parameter `IETYP` are given in Table II.3. Basically, first digit denotes the element structure (beam, shell, solid, shaft, solid to shell transition, solid to beam transition, user-defined), second digit is related to the order of the shape functions (linear, quadratic, cubic) used to define the element displacement field, and third digit describes the

element geometrical or intrinsic characteristics.

elem/elst

The various beam*, shell, solid, shaft, solid to shell transition and solid to beam transition elements are represented in Figure II.4. Note that these elements may be geometrically deformed (curved sides and edges) through a coordinate mapping calculated by the computer code. However, this transformation from a master element must satisfy some well-known basic rules, in order to ensure the positive definiteness of the Jacobian matrix associated with the mapping.

Note that the nodes of beam elements or the beam-type nodal points of solid to beam transition elements, as well as the associated auxiliary nodes, must lie in a plane. It should also be pointed out that rotating shaft elements may be geometrically deformed only in the local *r*-axis (unequally spaced nodes) and that all nodes of gyroscopic systems must be aligned with the rotation axis. Moreover, dynamic beam elements and rotating shaft elements may not be combined in the same mesh.

(3) The 3 digits correspond to the local (*r*, *s*, *t*) coordinate system of the element. Variable `INTEG` determines the numerical integration order (Gauss quadrature) used in the *r*-, *s*- and *t*-directions in the computation of the element stiffness matrix (see Table II.4).

For hermitian and dynamic beam elements, parameter `INTEG` is reset to 333. For isoparametric beam and cylindrical shaft elements, selective and reduced integration schemes lead to the same stiffness matrix. With shell, solid, shaft, solid to shell transition and solid to beam transition elements, the selective integration scheme is based on the so-called B-approach (reduced integration of the transverse shear strains). For incomplete quadrangular shell elements (232, 242), the number of integration points in the *r*- or *s*-direction is defined according to the maximum number of nodes located on one of the two element edges directed towards the *r*- or *s*-axis respectively. For the incomplete brick solid element (332), the number of integration points in the *r*-, *s*- or *t*-direction is defined according to the maximum number of nodal points located on one of the four element edges directed towards the *r*-, *s*- or *t*-axis respectively. For triangular shell elements (223, 233, 243), prismatic solid elements (323, 333) and prismatic solid to shell transition elements (523, 533, 534, 535, 538, 539), selective or reduced integration is used as soon as one of the first two digits of parameter `INTEG` is equal to 2 or 1 respectively. With tetrahedric solid elements (324, 334), selective or reduced integration is applied as soon as one of the three digits of parameter `INTEG` is equal to 2 or 1 respectively. For shaft elements, the stiffness matrix terms are exactly integrated in the transverse directions of the element.

The consistent mass matrix (all element types) and the gyroscopic matrix (rotating shaft elements) are always evaluated using the full integration scheme. Note that for hermitian and dynamic beams, the row-sum technique and the special lumping scheme lead to the same diagonal mass matrix (see dataset `PROB/MASS`). This is also the case for isoparametric beam, shell, solid, shaft, solid to shell transition and solid to beam transition elements when their nodal points are equally spaced, their geometry regular (straight line, rectangle or brick) and their nodal thicknesses (shell and solid to shell transition elements) or section properties

* Truss elements are defined as beam elements (neglecting shear deformation effects) with zero bending and torsional moments of inertia (see dataset `ELEM/GEOM`)

(beam, shaft and solid to beam transition elements) constant throughout the element.

elem/elst

(4) For each element set must be related a material property set defined on dataset `elem/MATL`. Note that a material property set can be associated with more than one element set.

Table II.3 Element types available in the computer code.

Element description	Element type	Number of nodes
Linear isoparametric beam element	120	2
Hermitian beam element (shear effects neglected)*	121	2
Hermitian beam element (shear effects included)	122	2
Dynamic beam element (shear effects neglected)*	123	2
Dynamic beam element (shear effects included)	124	2
Quadratic isoparametric beam element	130	3
Cubic isoparametric beam element	140	4
Bilinear quadrangular shell element	220	4
Bilinear triangular shell element	223	3
Biquadratic lagrangian quadrangular shell element	230	9
Quadratic serendipian quadrangular shell element	231	8
Incomplete quadratic serendipian quadrangular shell element	232	5–8
Quadratic triangular shell element	233	4–6
Bicubic lagrangian quadrangular shell element	240	16
Cubic serendipian quadrangular shell element	241	12
Incomplete cubic serendipian quadrangular shell element	242	6–12
Cubic triangular shell element	243	5–10
Trilinear brick solid element	320	8
Linear prismatic solid element	323	6
Linear tetrahedric solid element	324	4
Triquadratic lagrangian brick solid element	330	27
Quadratic serendipian brick solid element	331	20
Incomplete quadratic serendipian brick solid element	332	9–20
Quadratic prismatic solid element (complete or incomplete)	333	7–18
Quadratic tetrahedric solid element (complete or incomplete)	334	5–10
Biquadratic-linear lagrangian brick solid element	335	18
Incomplete triquadratic lagrangian brick solid element	336	21–26

(cont.)

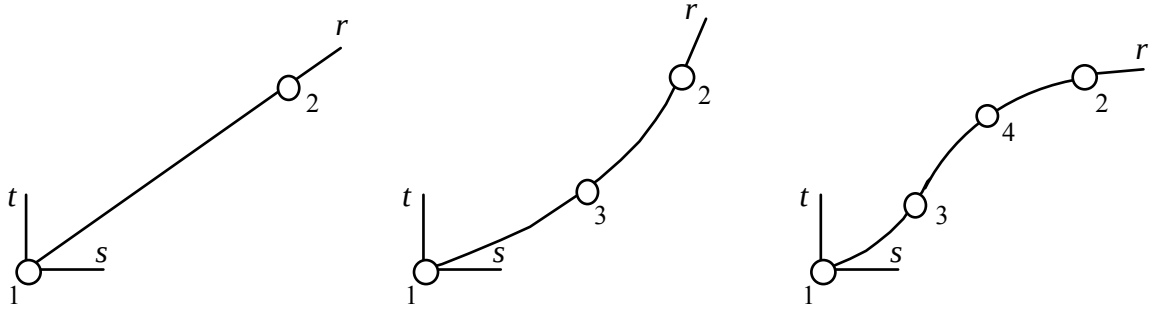
* Element type valid for truss elements

Table II.3 *(cont.)* Element types available in the computer code.

Element description	Element type	Number of nodes*
Linear isoparametric shaft element (cylindrical shape)	420	2
Linear isoparametric shaft element (tapered shape)	421	2
Quadratic isoparametric shaft element (cylindrical shape)	430	3
Quadratic isoparametric shaft element (tapered shape)	431	3
Cubic isoparametric shaft element (cylindrical shape)	440	4
Cubic isoparametric shaft element (tapered shape)	441	4
Linear brick solid to shell transition element	520	6 (4)
Linear prismatic solid to shell transition element	523	5 (4)
Linear brick solid to shell transition element (line interface)	526	5 (2)
Linear prismatic solid to shell transition element (line interface)	528	4 (2)
Quadratic lagrangian brick solid to shell transition element	530	15 (9)
Quadratic serendipian brick solid to shell transition element	531	14 (8)
Biquadratic-linear brick solid to shell transition element	532	12 (6)
Quadratic lagrangian prismatic solid to shell transition element	533	12 (9)
Quadratic serendipian prismatic solid to shell transition element	534	11 (8)
Biquadratic-linear prismatic solid to shell transition element	535	9 (6)
Quadratic brick solid to shell transition element (line interface)	536	11 (3)
Biquadratic-linear brick solid to shell transition element (line interface)	537	10 (2)
Quadratic prismatic solid to shell transition element (line interface)	538	8 (3)
Biquadratic-linear prismatic solid to shell transition element (line interface)	539	7 (2)
Linear solid to beam transition element	620	5 (4)
Quadratic lagrangian solid to beam transition element	630	11 (9)
Quadratic serendipian solid to beam transition element	631	10 (8)
Biquadratic-linear solid to beam transition element	632	8 (6)
Cubic-quadratic lagrangian solid to beam transition element	640	12 (9)
Cubic-quadratic serendipian solid to beam transition element	641	11 (8)
Cubic-quadratic-linear solid to beam transition element	642	9 (6)
User-defined element	900	–

* Bracketed values are referred to as the number of solid-type nodal points

elem/elst

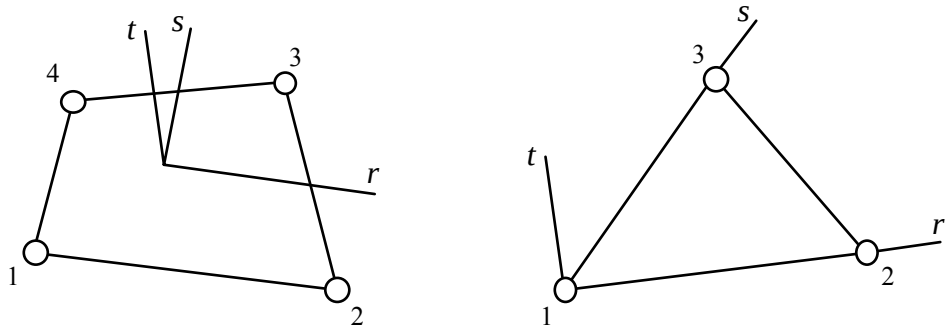


120 (2)
121 (2)
122 (2)
123 (2)
124 (2)

130 (3)

140 (4)

Figure II.4a Hermitian, isoparametric and dynamic beam elements (Bracketed values are referred to as the number of nodes).



220 (4)

223 (3)

Figure II.4b Linear superparametric shell elements (Bracketed values are referred to as the number of nodes).

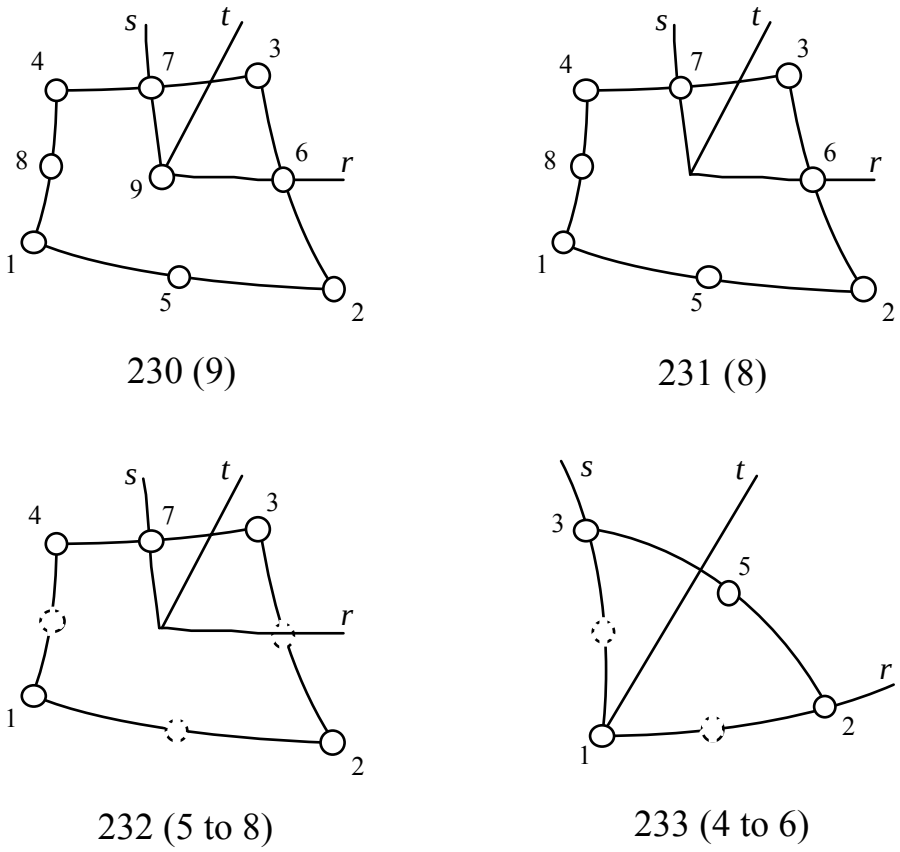


Figure II.4c Quadratic superparametric shell elements (Bracketed values are referred to as the number of nodes).

elem/elst

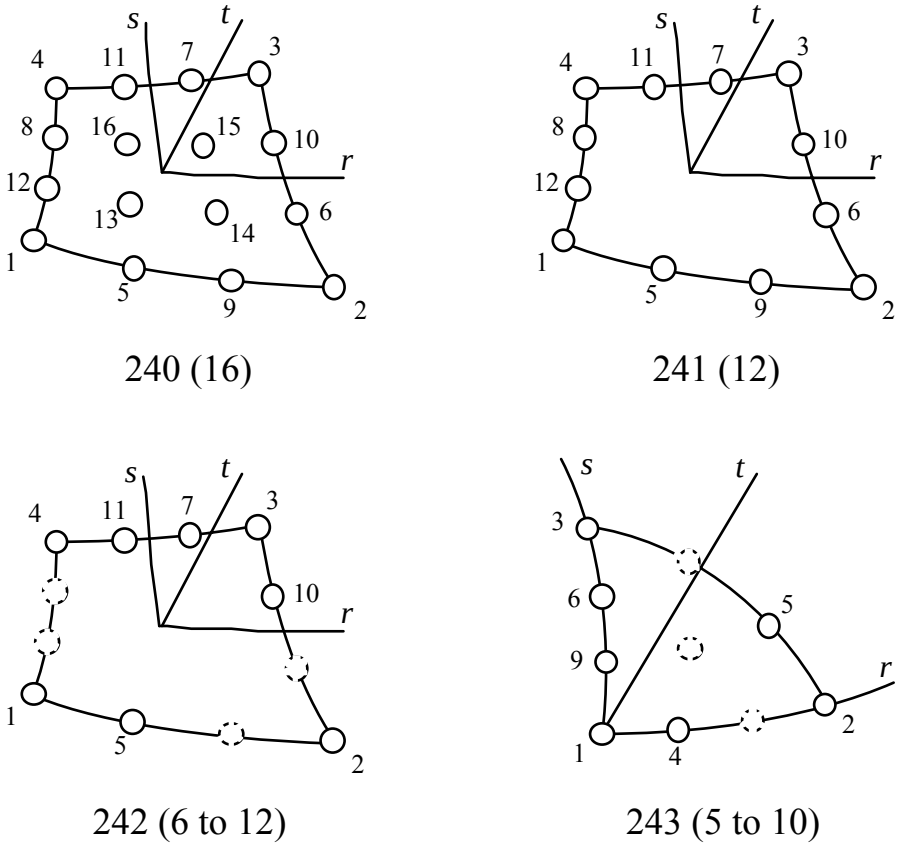
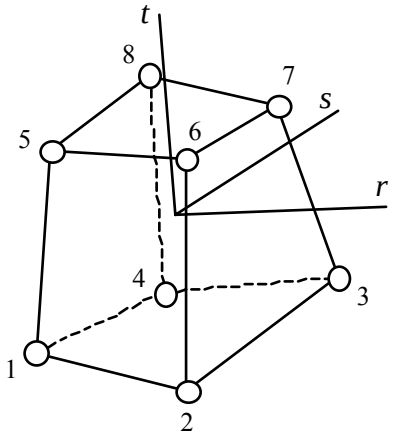
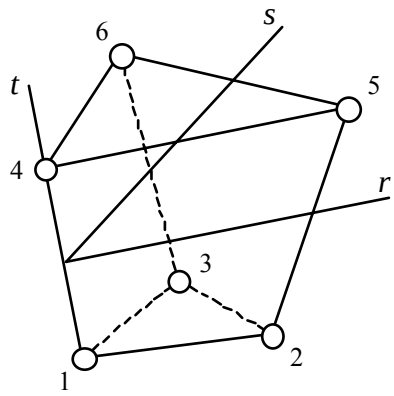


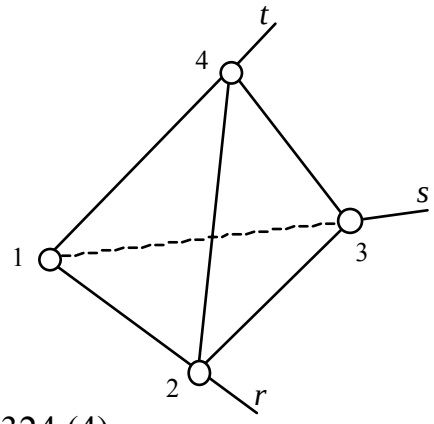
Figure II.4d Cubic superparametric shell elements (Bracketed values are referred to as the number of nodes).



320 (8)



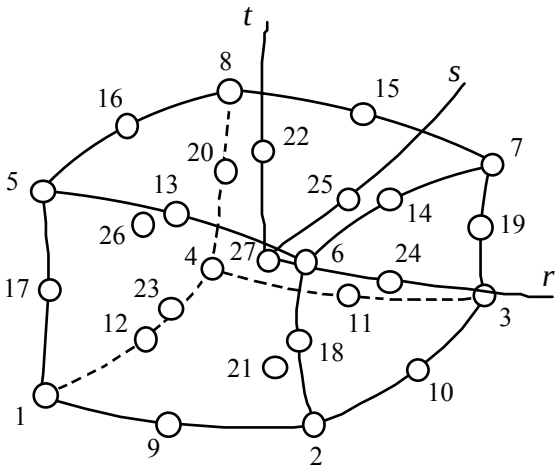
323 (6)



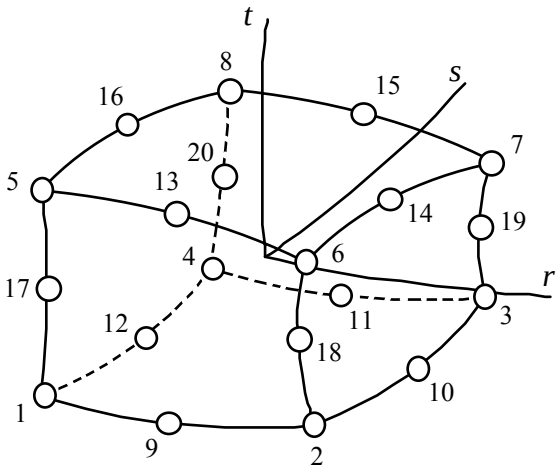
324 (4)

Figure II.4e Linear isoparametric solid elements (Bracketed values are referred to as the number of nodes).

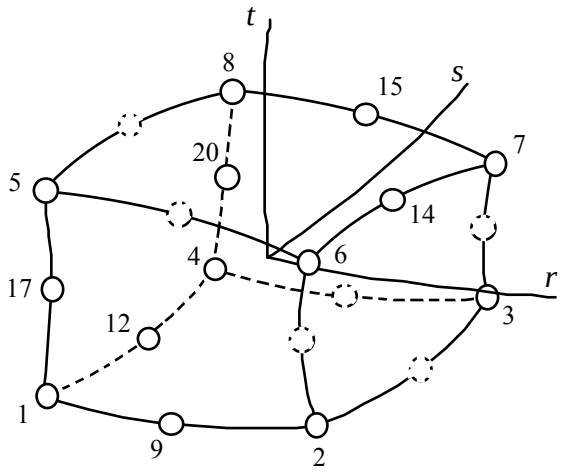
elem/elst



330 (27)



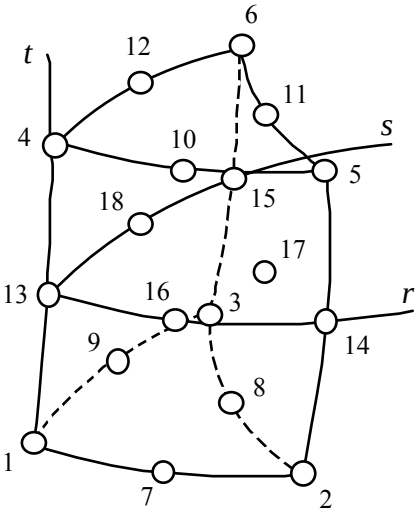
331 (20)

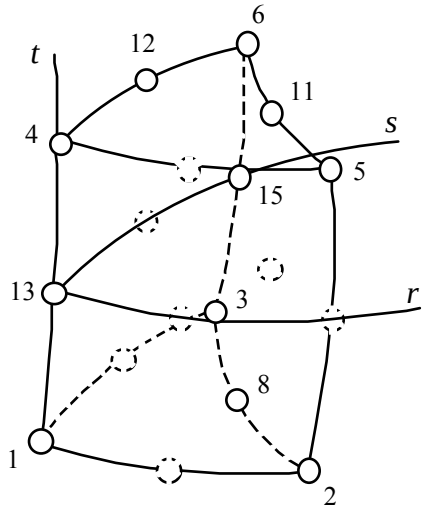


332 (9 to 20)

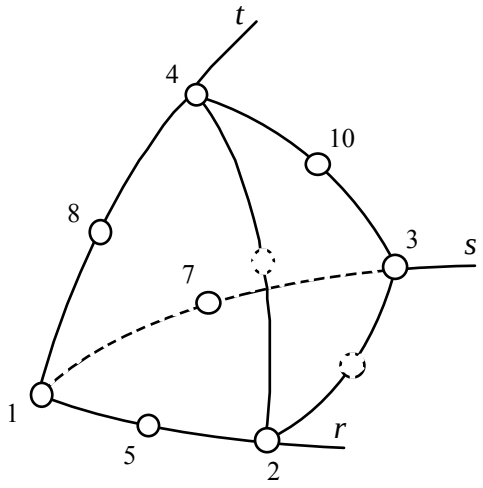
(cont.)

Figure II.4f Quadratic isoparametric solid elements (Bracketed values are referred to as the number of nodes).

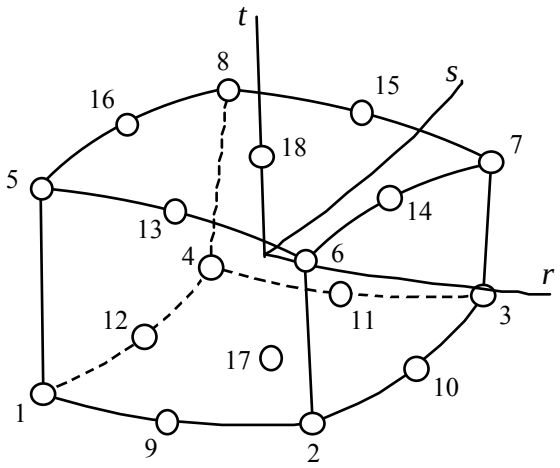




333 (7 to 18)



334 (5 to 10)

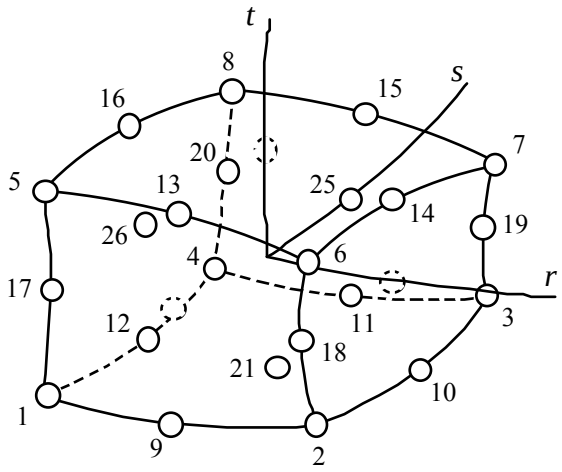


335 (18)

(cont.)

Figure II.4f *(cont.)* Quadratic isoparametric solid elements (Bracketed values are referred to as the number of nodes).

elem/elst



336 (21 to 26)

Figure II.4f *(cont.)* Quadratic isoparametric solid elements (Bracketed values are referred to as the number of nodes).

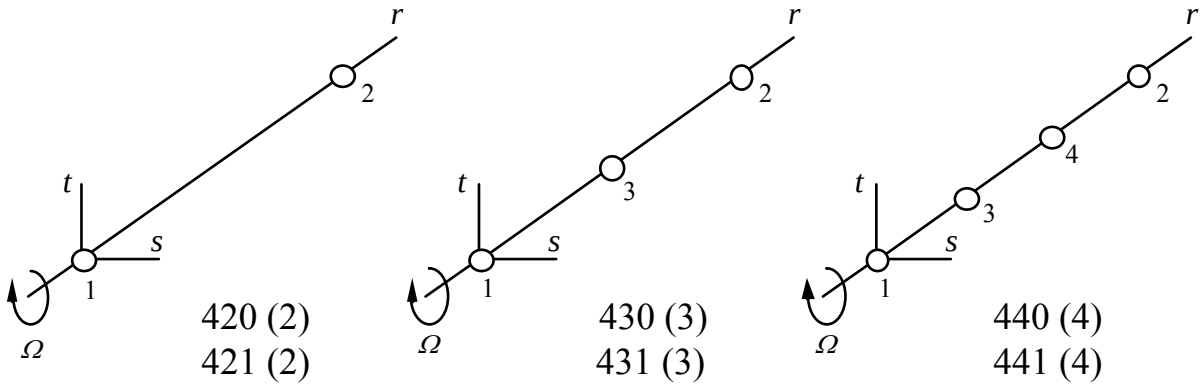
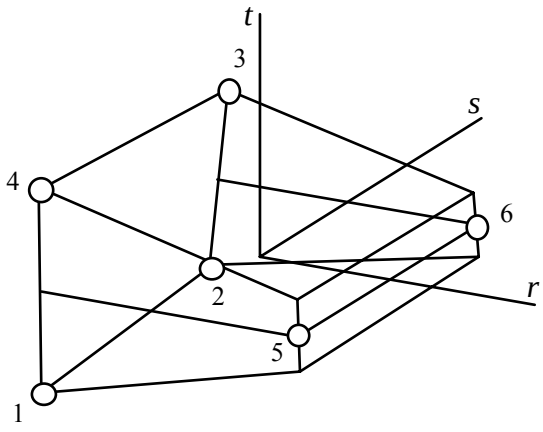
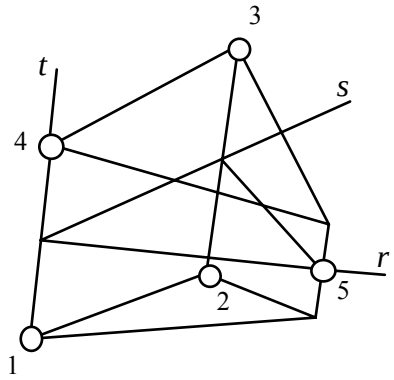


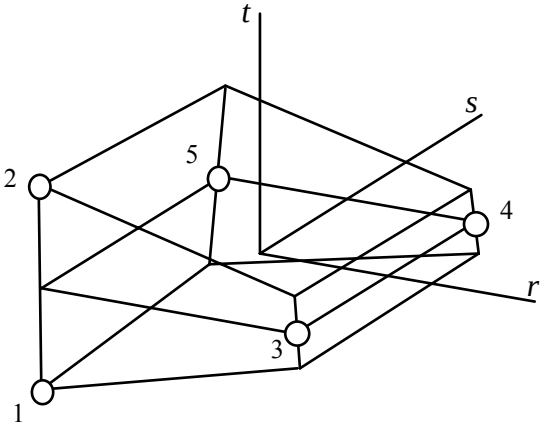
Figure II.4g Isoparametric rotating shaft elements (Bracketed values are referred to as the number of nodes).



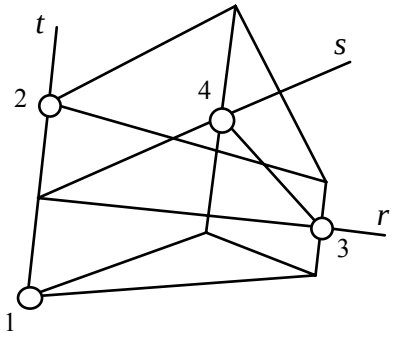
520 (6)



523 (5)



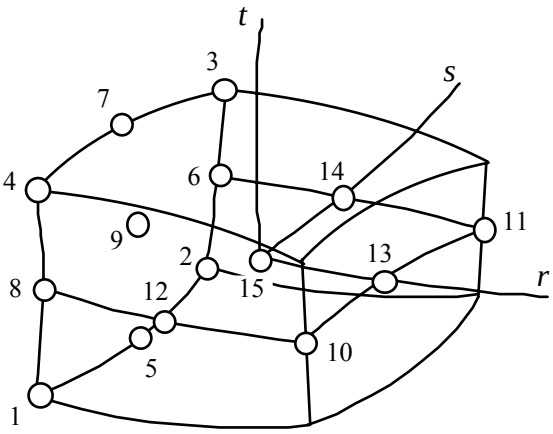
526 (5)



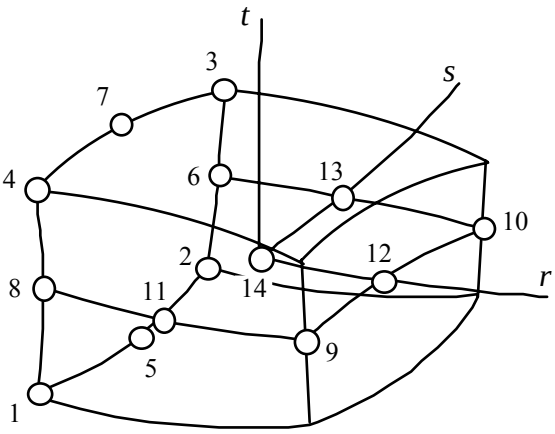
528 (4)

Figure II.4h Linear solid to shell transition elements (Bracketed values are referred to as the number of nodes).

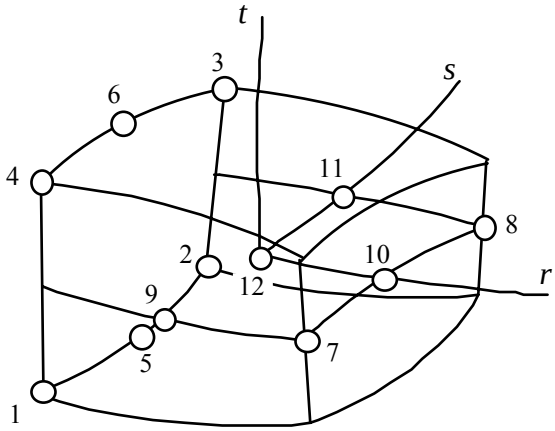
elem/elst



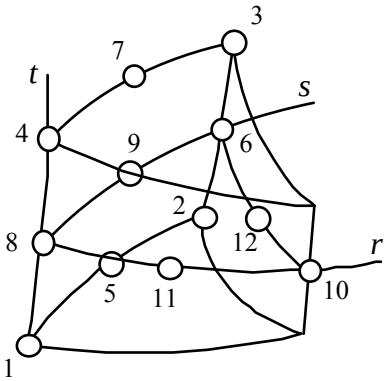
530 (15)



531 (14)



532 (12)

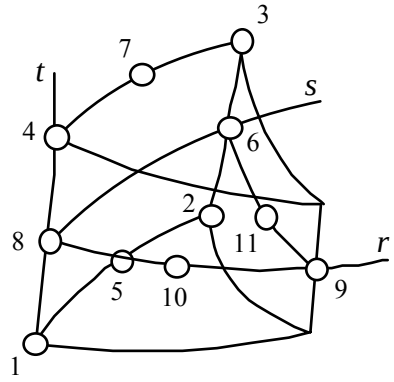


533 (12)

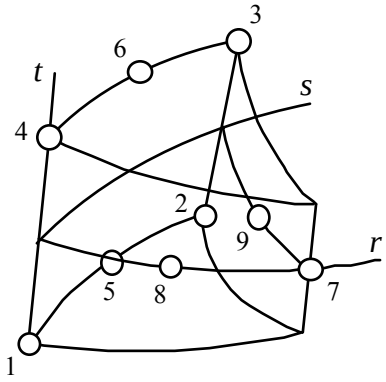
(cont.)

Figure II.4i Quadratic solid to shell transition elements (Bracketed values are referred to as the number of nodes).

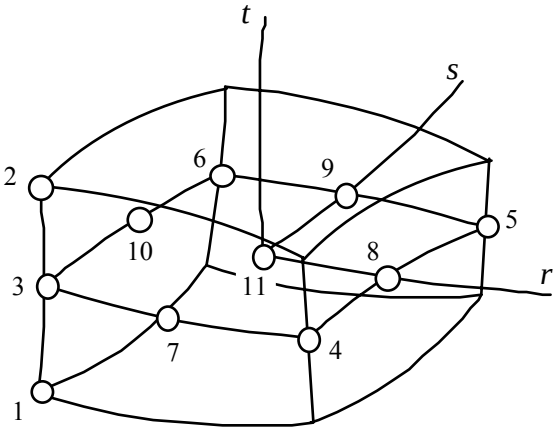
elem/elst



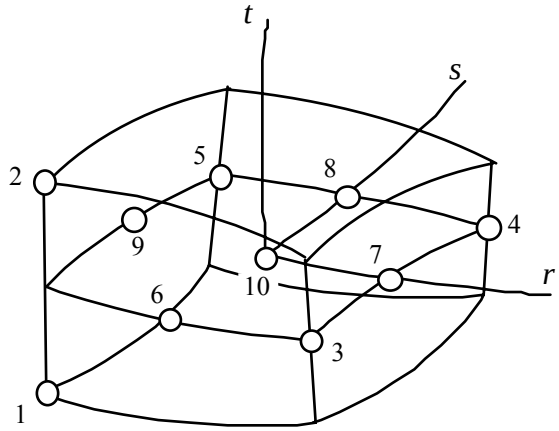
534 (11)



535 (9)



536 (11)

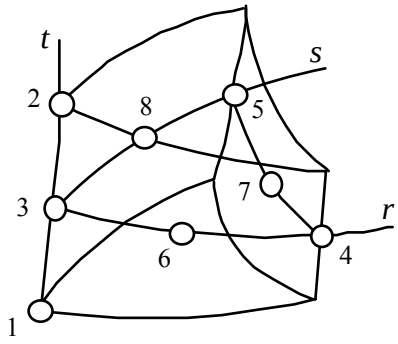


537 (10)

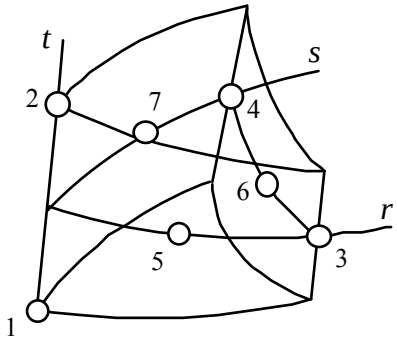
(cont.)

Figure II.4i *(cont.)* Quadratic solid to shell transition elements (Bracketed values are referred to as the number of nodes).

elem/elst

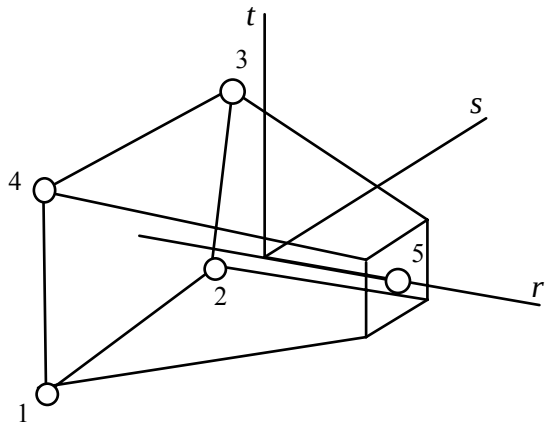


538 (8)



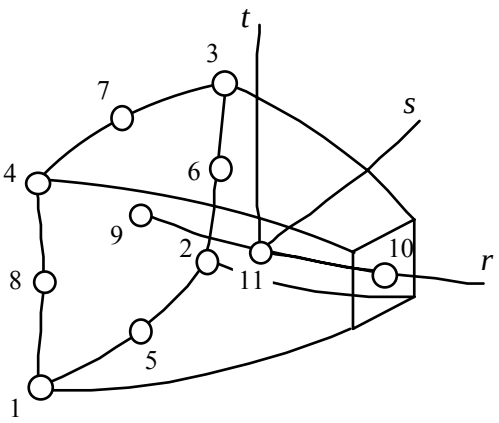
539 (7)

Figure II.4i *(cont.)* Quadratic solid to shell transition elements (Bracketed values are referred to as the number of nodes).

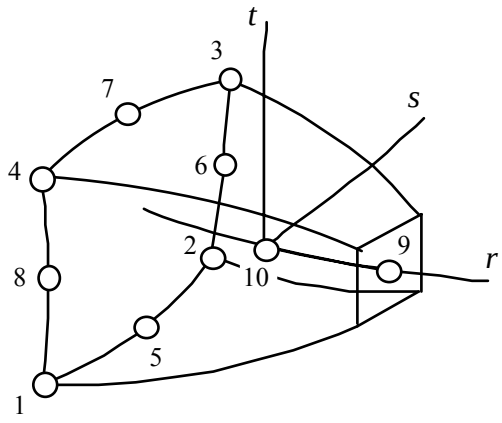


620 (5)

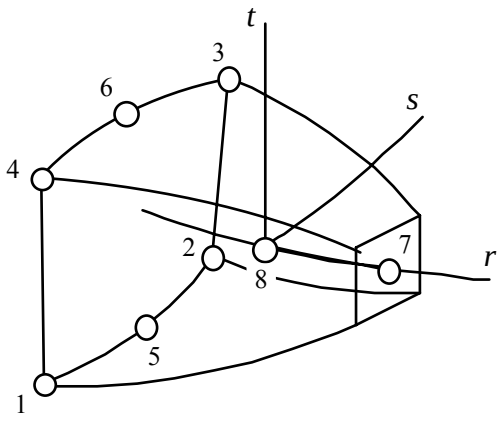
Figure II.4j Linear solid to beam transition elements (Bracketed values are referred to as the number of nodes).



630 (11)



631 (10)



632 (8)

Figure II.4k Quadratic solid to beam transition elements (Bracketed values are referred to as the number of nodes).

elem/elst

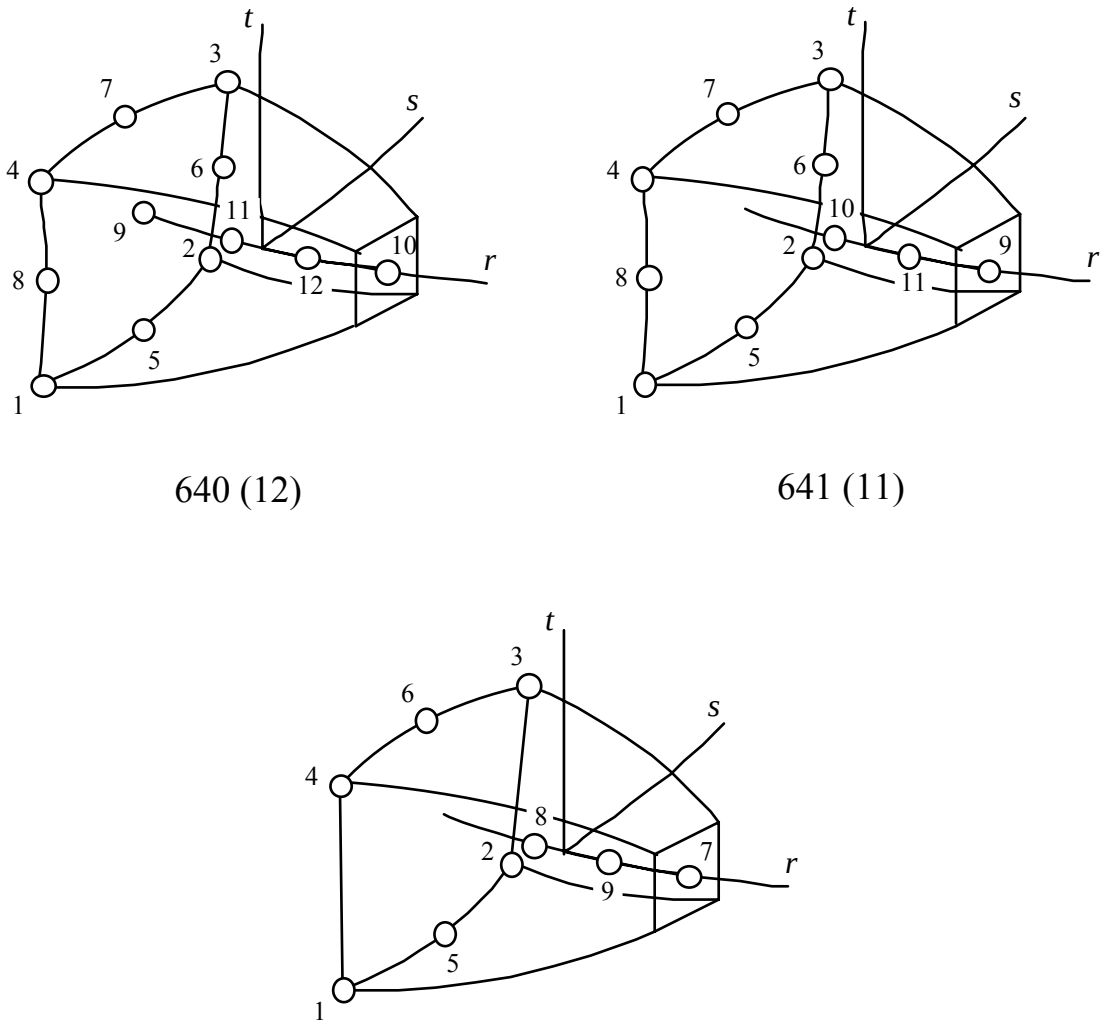


Figure II.4l Cubic solid to beam transition elements (Bracketed values are referred to as the number of nodes).

Table II.4 Number of Gauss integration points.

Element type	Full integration			Reduced integration		
	r	*s*	*t*	*r*	*s*	*t*
120	2	4	4	1	3	3
121	---------exact---------			---------exact---------		
122	---------exact---------			---------exact---------		
123	---------exact---------			---------exact---------		
124	---------exact---------			---------exact---------		
130	3	4	4	2	3	3
140	4	4	4	3	3	3
220	2	2	2	1	1	1
223	------4------		2	------1------		1
230	3	3	2	2	2	1
231	3	3	2	2	2	1
232	2 or 3	2 or 3	2	1 or 2	1 or 2	1
233	------7------		2	------4------		1
240	4	4	2	3	3	1
241	4	4	2	3	3	1
242	2 to 4	2 to 4	2	1 to 3	1 to 3	1
243	------13------		2	------7------		1
320	2	2	2	1	1	1
323	------4------		2	------1------		1
324	----------5----------			----------1----------		
330	3	3	3	2	2	2
331	3	3	3	2	2	2
332	2 or 3	2 or 3	2 or 3	1 or 2	1 or 2	1 or 2
333	------7------		3	------4------		2
334	----------17----------			----------1----------		
335	3	3	2	2	2	1
336	3	3	3	2	2	2
420	2	---exact---		1	---exact---	
421	4	---exact---		2	---exact---	
430	3	---exact---		2	---exact---	
431	5	---exact---		3	---exact---	
440	4	---exact---		3	---exact---	
441	6	---exact---		4	---exact---	

(cont.)

elem/elst

Table II.4 *(cont.)* Number of Gauss integration points.

Element type	Full integration			Reduced integration		
	r	*s*	*t*	*r*	*s*	*t*
520	2	2	2	1	1	1
523	------4------		2	------1------		1
526	2	2	2	1	1	1
528	------4------		2	------1------		1
530	3	3	3	2	2	2
531	3	3	3	2	2	2
532	3	3	2	2	2	1
533	------7------		3	------4------		2
534	------7------		3	------4------		2
535	------7------		2	------4------		1
536	3	3	3	2	2	2
537	3	3	2	2	2	1
538	------7------		3	------4------		2
539	------7------		2	------4------		1
620	2	2	2	1	1	1
630	3	3	3	2	2	2
631	3	3	3	2	2	2
632	3	3	2	2	2	1
640	4	3	3	3	2	2
641	4	3	3	3	2	2
642	4	3	2	3	2	1
900	–	–	–	–	–	–

II.4.2 ***Material properties*** (I5, 4F10.0, I5)

```
MATL
IMATL YOUNG POISS DENST SHEAR ISSLW
```

Variable	Columns	Value	Note	Description
IMATL	1–5	[1,NMATL]	(1)	Material property set number
YOUNG	6–15	]0.,∞[	(–)	Young's modulus
POISS	16–25	]0.,0.5[	(–)	Poisson's ratio
DENST	26–35	]0.,∞[	(2)	Mass density
SHEAR	36–45	]0.,∞[	(3)	Shear factor (Default : 1.)
ISSLW	46–50	[0,3]	(4)	Stress-strain law type 0 not applicable 1 mixed formulation 2 beam- or shell-type 3 solid-type

(1) Material property sets must be input in ascending order and may not be generated.

(2) The mass density is used directly in the computation of the element mass and possibly gyroscopic matrices (no acceleration constants are applied to variable DENST). The mass density is also employed for the calculation of the load vector if ground acceleration is defined in section ALGO/MSUP/INTE.

(3) The shear factor is only used for isoparametric beam elements, superparametric shell elements and solid to shell or solid to beam transition elements. This factor improves the shear deformation characteristics of thick beams or shells and thick beam-like or shell-like transition regions; it may appropriately be set equal to 0.833. Note that the shear effects are always included in solid and rotating shaft elements. For hermitian or dynamic beam elements including shear effects, the shear factor is implicitly computed by means of the effective shear areas defined on dataset elem/ GEOM.

(4) The stress-strain law type must be specified only when solid to shell or solid to beam transition elements are associated with the current material property set. Note that variable ISSLW is ignored by the code for elements of another type, even if these elements are associated with the same material property set as the one used for transition elements.

elem/matl

Parameter `ISSLW` defines the choice of the generalized stress-strain matrix to be used for the computation of the stiffness matrix. Note that, if the transition element is used only for a proper adjustment of the number of degrees of freedom, i. e. no geometrical transition has to be modelled, the selection of the elasticity matrix should be governed by the localization of the element in the mesh. That is to say, if the element is situated at the limit of the beam- or shell-like portion of the structure (the geometrical transition being realized with solid elements), the beam- or shell-type stress-strain law should be applied and vice versa.

In the case commonly encountered where the element must model, at least partially, the geometrical transition region, a mixed formulation has to be adopted since the stress-strain state within the element constitutes a smooth transition from the three-dimensional to the beam or shell stress-strain state when moving from the solid interface toward the beam- or shell-type one. In the computer code, this approach is applied by using the solid-type stress-strain matrix for the integration points located in the vicinity of the connected solid elements and the beam- or shell-type elasticity matrix is applied elsewhere.

II.4.3 *Geometrical properties of beam (or truss) elements* (*optional* / I5, 6F10.0)

Skip this section if no beam (or truss) element set is defined on dataset `elem/ELST`.

```
GEOM
IELST TORSR BENDS BENDT AREAR AREAS AREAT
```

Variable	Columns	Value	Note	Description
`IELST`	1–5	[1,`NELST`]	(1)	Beam element set number
`TORSR`	6–15	[0.,∞[	(2)	Torsional moment of inertia about local *r*-axis
`BENDS`	16–25	[0.,∞[	(2)	Bending moment of inertia about local *s*-axis
`BENDT`	26–35	[0.,∞[	(2)	Bending moment of inertia about local *t*-axis
`AREAR`	36–45	]0.,∞[	(3)	Normal cross-sectional area
`AREAS`	46–55	[0.,∞[	(3)	Effective shear area in the *s*-direction
`AREAT`	56–65	[0.,∞[	(3)	Effective shear area in the *t*-direction

(1) Geometrical properties are input in ascending order for element sets associated with hermitian, isoparametric or dynamic beam (and truss) elements.

(2) The section moments of inertia are the principal moments of inertia in the local (*r*, *s*, *t*) coordinate system of the beam element. For isoparametric beam elements, the cross-section is rectangular and only parameters `BENDS` and `BENDT` are used in this dataset. Furthermore, for this type of element, the bending moments of inertia must be positive definite. For truss elements, variables `TORSR`, `BENDS` and `BENDT` must be set to zero.

(3) If the effective shear area `AREAS` or `AREAT` is set equal to zero, the shear deformation in the *s*- or *t*-direction respectively is neglected.

elem/sect

II.4.4 *Section properties of rotating shaft elements* (*optional* / I5, 4F10.0)

Skip this section if no shaft element set is defined on dataset `elem/ELST`.

```
SECT
IELST DMEXL DMINL DMEXR DMINR
```

Variable	Columns	Value	Note	Description
IELST	1–5	[1,NELST]	(1)	Shaft element set number
DMEXL	6–15	]0.,∞[	(2)	External diameter at local node 1 (left end) of first element in current set
DMINL	16–25	[0.,DMEXL[	(2)	Internal diameter at local node 1 (left end) of first element in current set
DMEXR	26–35	]0.,∞[	(2)	External diameter at local node 2 (right end) of last element in current set
DMINR	36–45	[0.,DMEXR[	(2)	Internal diameter at local node 2 (right end) of last element in current set

(1) Section properties are input in ascending order for element sets associated with rotating shaft elements.

(2) For cylindrical shaft elements, parameters DMEXR and DMINR are not used and all elements in current set are assumed to have the same section properties, i. e. DMEXL as external diameter and DMINL as internal diameter. For tapered elements, the external and internal diameters are determined by linearly interpolating the section properties defined for the left and right ends of the element set. Note that for solid shaft elements, DMINL and/or DMINR must be set to zero.

elem/gyro

II.4.5 *Angular speed information for rotating shaft elements* (*optional* / 3F10.0)

Skip this section if no shaft element set is defined on dataset `elem/ELST`.

```
GYRO
RPMIN RPMAX RPINC
```

Variable	Columns	Value	Note	Description
`RPMIN`	1–10	[0.,∞[	(1)	Initial angular speed (Unit : rpm or (60 unit-time)$^{-1}$)
`RPMAX`	11–20	[`RPMIN`,∞[	(1)	Final angular speed (Unit : rpm or (60 unit-time)$^{-1}$)
`RPINC`	21–30	[0.,∞[	(2)	Angular speed increment (Default : see note, unit : rpm or (60 unit-time)$^{-1}$)

(1) The angular speed information is used to determine the components of the skew-symmetric gyroscopic matrix of the finite element assemblage. Variables `RPMIN`, `RPMAX` and `RPINC` characterize the spin rates at which an eigensolution must be extracted. Note that the eigenpairs of the system at rest (zero spin rate) are always computed, even if parameter `RPMIN` is set to a non-zero value.

(2) Parameter `RPINC` must be an integer multiple of the difference between the starting and final spin rates. By default, the value of `RPINC` is set to that difference.

`elem/dime`

II.4.6 *Cross-sectional dimensions of solid to beam transition elements*

(*optional* / I5, 6F10.0)

Skip this section if no solid to beam transition element set is defined on dataset `elem/ELST`.

```
DIME
IELST DIMS1 DIMT1 DIMS2 DIMT2 DIMS3 DIMT3
```

Variable	Columns	Value	Note	Description
`IELST`	1–5	[1,`NELST`]	(1)	Solid to beam transition element set number
`DIMS1`	6–15	]0.,∞[	(2)	Cross-sectional dimension in the *s*-direction at first beam-type nodal point
`DIMT1`	16–25	]0.,∞[	(2)	Cross-sectional dimension in the *t*-direction at first beam-type nodal point
`DIMS2`	26–35	]0.,∞[	(2)	Cross-sectional dimension in the *s*-direction at second beam-type nodal point
`DIMT2`	36–45	]0.,∞[	(2)	Cross-sectional dimension in the *t*-direction at second beam-type nodal point
`DIMS3`	46–55	]0.,∞[	(2)	Cross-sectional dimension in the *s*-direction at third beam-type nodal point
`DIMT3`	56–65	]0.,∞[	(2)	Cross-sectional dimension in the *t*-direction at third beam-type nodal point

(1) Cross-sectional dimensions are input in ascending order for element sets associated with solid to beam transition elements.

(2) The input order of the cross-sectional dimensions (by pairs of values) corresponds to the conventional order sequence used for the numbering of the beam-type nodal points of the solid to beam transition elements (end node first, followed by the first and second internal nodes if necessary). See also Figures II.4j to II.4l.

II.4.7 *Element connectivities* (*repeatable*)

The element connectivity sets must be input in the element set sequence defined on dataset `elem/ELST`.

II.4.7.1 *Beam (or truss) element connectivities* (5I5 / 5I5)

Skip this section if no beam (or truss) element set is defined on dataset `elem/ELST`.

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
ANODE NODE1 NODE2 NODE3 NODE4
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[2,4]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,1]	(4)	Generation option 0 no generation 1 1-D block generation (line)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`ANODE`	1–5	[1,`NNODE`]	(5)	Auxiliary node in (*r*-*s*) local plane

(cont.)

elem/cntv

(cont.)

Variable	Columns	Value	Note	Description
NODE1	6–10	[1,NNODE]	(6)	Global node number of element nodal point 1
…				
NODE4	21–25	[1,NNODE]	(6)	Global node number of element nodal point 4

(1) First element of set must be element number 1; last element, corresponding to element number Nelem of current set (see section elem/ELST), must be input and may not be generated. Other elements may be defined in any order sequence.

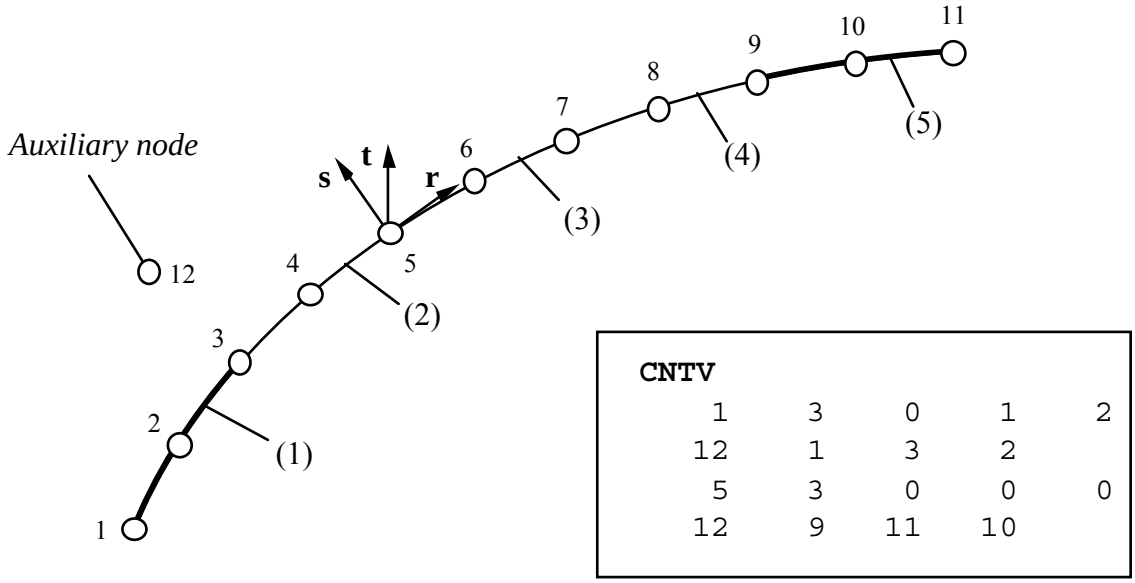
(2) The number of nodes used to describe beam elements (hermitian, isoparametric or dynamic type) and truss elements is shown in Table II.3. By default, the values given in that table are used. See also Figure II.4a.

(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since local stiffness and mass matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) Two elements must be defined for line generation (Fig. II.5). The number of generated elements, as well as their numbering, is determined according to the global nodal point increment used to compute node numbers of missing elements. Note that parameter NSTEP must be compatible with the generation scheme and that the nodes may be generated increasingly (NSTEP positive) or decreasingly (NSTEP negative). Second element in the generation set may be used as first element of a new generation set. If the similarity option is active, the stiffness and mass matrices of the generated elements are assumed to be identical to those of the first element of the generation set.

(5) The auxiliary node is merely used to define the (*r*-*s*) plane of the element, i.e. the directions of the principal axes, which are necessary to define the geometrical properties of the element (see section elem/GEOM). Note that any appropriate node that is defined for the elements can be chosen as auxiliary nodal point. If an additional node out of the element assemblage is defined, all degrees of freedom associated to that node must be deleted using the nodal boundary condition code ICODE on dataset NODE/BCND. If the line generator is active, the auxiliary nodes associated to the generated elements are interpolated from the auxiliary nodal points of the two elements defined in the generation set; however, the same auxiliary node can be used for all the generated elements.

(6) The input sequence that must be followed by element node input is given in Figure II.4a. Only the first `NNOEL` global node numbers are read. It should be noted that all the nodes (auxiliary node included) of the isoparametric beam element must initially lie in the (*r*-*s*) plane. If the line generation option is used, the node numbering of the missing elements is determined by interpolating, with reference to parameter `NSTEP`, the global nodal point numbers of the two elements in the generation set.




```
CNTV
     1     3     0     1     2
    12     1     3     2
     5     3     0     0     0
    12     9    11    10
```

Figure II.5 Beam element generation along a line (**r** tangent to the centroidal axis, **t** cross-product of **r** and the vector from nodal point `NODE1` to the auxiliary node `ANODE`, **s** cross-product of **t** and **r**)

elem/cntv

II.4.7.2 *Superparametric shell element connectivities* (5I5 / 16I5)

Skip this section if no shell element set is defined on dataset `elem/ELST`. *See also section* II.4.8 *for shell element connectivity generation (special block generator).*

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
NODE1 NODE2 NODE3 NODE4 NODE5 NODE6 ...            NOD16
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[3,16]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,2]	(4)	Generation option 0 no generation 1 1-D block generation (line) 2 2-D block generation (surface)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`NODE1`	1–5	[1,`NNODE`]	(5)	Global node number of element nodal point 1
…				
`NOD16`	76–80	[1,`NNODE`]	(5)	Global node number of element nodal point 16

(1) First element of set must be element number 1; last element, corresponding to element number `Nelem` of current set (see section `elem/ELST`), must be input and may not be generated. Other elements may be defined in any order sequence.

(2) The number of nodes used to describe superparametric shell elements is shown in Table II.3. By default, the values given in that table are used. For incomplete quadratic or cubic quadrangular and triangular elements, the value corresponding to the complete element is selected by default. See also Figures II.4b to II.4d.

(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since local stiffness and mass matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) The 1-D block generation scheme is identical to the one used for the generation of beam elements along a line (see note 4 in section II.4.7.1).

For 2-D block generation over a surface (Fig. II.6), the generation set must contain information related to 3 elements. The number of generated elements, as well as their numbering in the two generation directions, are determined according to the global node increments (parameter `NSTEP` specified for the first two elements in the generation set) used to compute the nodal point numbers of missing elements. Note that the variables `NNOEL` for the three elements in the generation set must be equal and that the parameters `IGENE` defined for the first two elements must be set to 2. Moreover, the values of variable `NSTEP` for the first two elements must be compatible with the generation scheme. Nodal points may be generated increasingly or decreasingly (`NSTEP` positive or negative) and the two directions selected for the element number generation must not necessarily coincide with the directions chosen for the node numbering (see Fig. II.6). Last element in the generation set may be used as first element of a new generation set. If the similarity option is active in the first and/or second generation directions (`ISAME` equal to 1 for the second and/or third elements of the generation set), the stiffness and mass matrices of the generated elements are assumed to be identical to those of the element(s) located in front of the first and/or second generation directions.

For complete quadrangular shell elements (220, 230, 231, 240, 241) in regular meshes, a simple and labor-saving block generator is available. See section II.4.8 for more information.

(5) The input sequence for the global node numbers assigned to the nodal points of the element is shown in Figures II.4b to II.4d. Node numbers must be input in a counter-clockwise order about the element *t*-axis. Note that this axis and all the mid-surface vectors at the element nodes must point towards the same side (top or bottom surface) of the element.

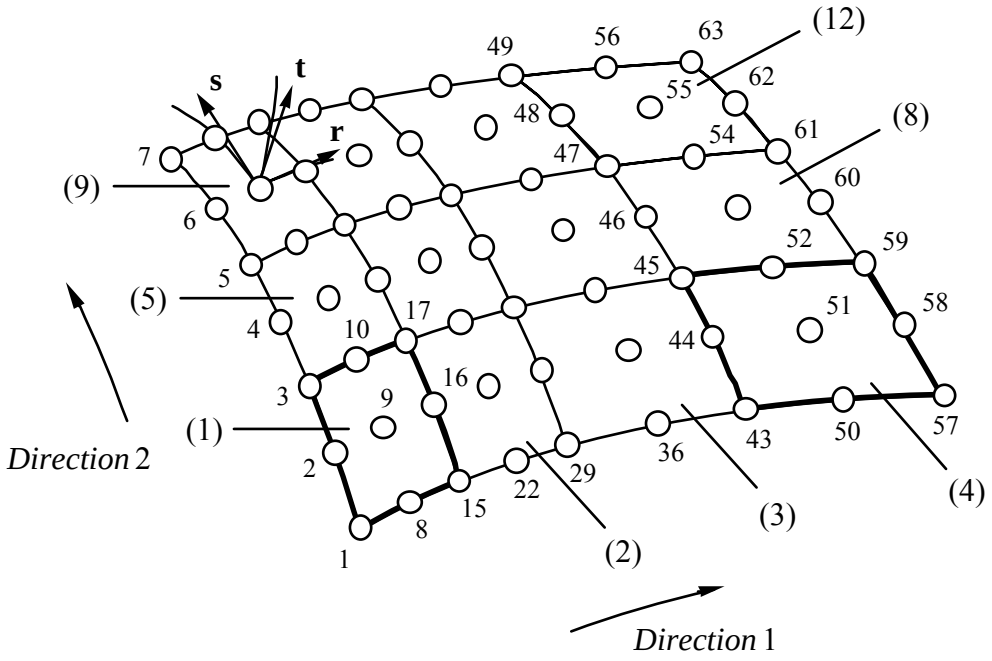
For linear elements (220, 223), and quadratic or cubic lagrangian (230, 240) and serendipian (231, 241) quadrangular elements, the first `NNOEL` global node numbers are read. For incomplete (232, 242) and quadratic or cubic triangular (233, 243) elements, a zero entry indicates that the node is not present; however, the total number of non-zero entries for these elements must be equal to `NNOEL`. For instance, the input sequence for the 8-node element 242 shown in Figure II.4d is

(1, 2, 3, 4, 5, 0, 7, 0, 0, 10, 11, 0). Similarly, the 7-node element 243 shown in the same figure is defined by the sequence (1, 2, 3, 4, 5, 6, 0, 0, 9, 0).

elem/cntv

Note that in the cubic triangular shell element, local node number 10 may be present only if the element is complete. Furthermore, for incomplete quadrangular elements, the four corner nodes must always be defined.

When the 1-D or 2-D generator is active, the node numbering of the missing elements is determined by interpolating, with reference to parameter `NSTEP`, the global nodal point numbers of the elements in the generation set.




```
CNTV
     1     9     0     2    14
     1    15    17     3     8    16    10     2     9
     4     9     0     2     2
    43    57    59    45    50    58    52    44    51
    12     9     0     0     0
    47    61    63    49    54    62    56    48    55
```

Figure II.6 Shell element generation over a surface (**r**, **s** tangent to the local coordinate axes, **t** normal to the mid-surface).

II.4.7.3 *Isoparametric solid element connectivities* (5I5 / 16I5 / 11I5)

Skip this section if no solid element set is defined on dataset `elem/ELST`. *See also section* II.4.8 *for solid element connectivity generation (special block generator)*

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
NODE1 NODE2 NODE3 NODE4 NODE5 NODE6 ...            NOD16
NOD17 NOD18 NOD19 NOD20 NOD21 NOD22 ...    NOD27
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[4,27]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,3]	(4)	Generation option 0 no generation 1 1-D block generation (line) 2 2-D block generation (surface) 3 3-D block generation (volume)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`NODE1`	1–5	[1,`NNODE`]	(5)	Global node number of element nodal point 1
…				
`NOD16`	76–80	[1,`NNODE`]	(5)	Global node number of element nodal point 16

(cont.)

elem/cntv

(cont.)

Variable	Columns	Value	Note	Description
NOD17	1–5	[1,NNODE]	(5)	Global node number of element nodal point 17
…				
NOD27	51–55	[1,NNODE]	(5)	Global node number of element nodal point 27

(1) First element of set must be element number 1; last element, corresponding to element number Nelem of current set (see section elem/ELST), must be input and may not be generated. Other elements may be defined in any order sequence.

(2) The number of nodes used to describe isoparametric solid elements is shown in Table II.3. By default, the values given in that table are used. For incomplete quadratic brick, prismatic and tetrahedric elements, the value corresponding to the complete element is selected by default. See also Figures II.4e and II.4f.

(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since local stiffness and mass matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) The 1-D and 2-D block generation schemes are similar to those developed for the generation of beam elements along a line and shell elements over a surface (see note 4 in sections II.4.7.1 and II.4.7.2).

For 3-D block generation within a volume (Fig. II.7), the generation set must contain information related to 4 elements. The number of generated elements, as well as their numbering in the three generation directions, are determined with reference to the global node increments (parameter NSTEP defined for the first three elements in the generation set) used to compute the nodal numbering of missing elements. Note that the variables NNOEL for the four elements in the generation set must be equal and that the parameters IGENE defined for the first three elements must be set to 3. Moreover, the values of variable NSTEP for the first three elements (node number increments in the three generation directions) must be compatible with the generation scheme. Nodal points may be generated increasingly or decreasingly (NSTEP positive or negative) and the three directions selected for the element number generation must not necessarily coincide with the directions chosen for the node numbering (see Figure II.7). Last element in the generation set may be used as first element of a new generation set. Note that, if the similarity option is active in the first, second and/or third generation directions (ISAME equal to 1 for the second, third and/or fourth elements of the generation set), the stiffness and mass matrices of the generated elements are assumed to be identical to those of the element(s) located in front of the first, second and/or third

generation directions (see Figure II.7).

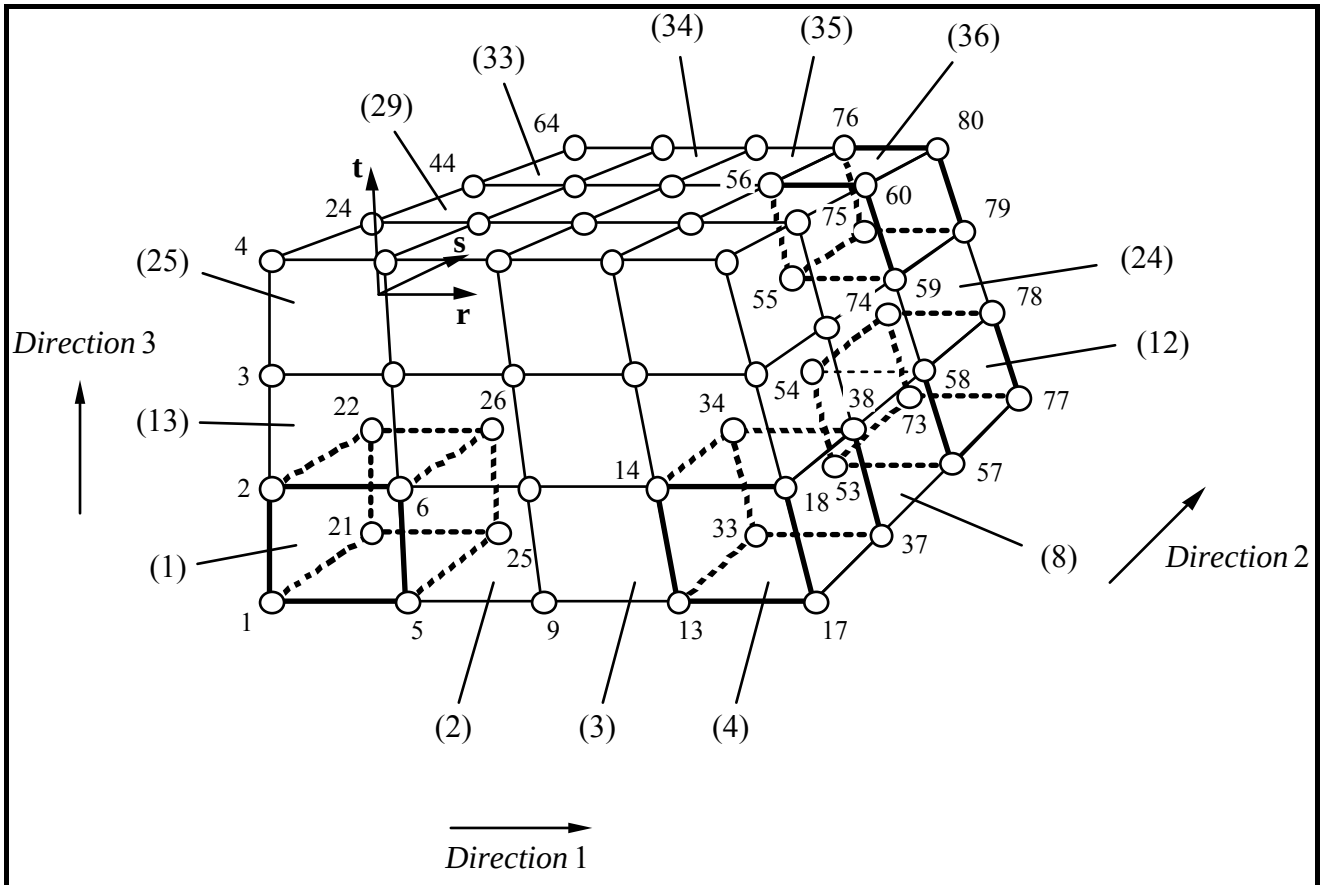
For complete brick solid elements (320, 330, 331) in regular meshes, a simple and labor-saving block generator is available. See section II.4.8 for more information.

(5) The input sequence for the global node numbers assigned to the nodal points of the element is shown in Figures II.4e and II.4f. For linear (320, 323, 324), quadratic-linear (335) and quadratic brick (330, 331) elements, the first `NNOEL` global node numbers are read. For quadratic incomplete brick elements (332), and quadratic prismatic (333) or tetrahedric (334) elements, a zero entry indicates that the node is not present in the element; nevertheless, the total number of non-zero entries for these elements must be equal to `NNOEL`. For example, the input sequence for the 14-node element 332 shown in Figure II.4f should be (1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 0, 12, 0, 14, 15, 0, 17, 0, 0, 20). Similarly, the 11-node element 333 depicted in the same figure is defined by the sequence (1, 2, 3, 4, 5, 6, 0, 8, 0, 0, 11, 12, 13, 0, 15, 0, 0, 0).

Note that in the quadratic prismatic solid element, local nodes 16, 17 and 18 may be present only if the other 15 nodal points are defined. Furthermore, for incomplete elements, the 8 corner nodes must always be specified. If the total number of entries (zero and non-zero entries) is less than or equal to 16, second global node number set introduced for `NOD17` to `NOD27` must be omitted.

When the generators are active, the node numbering of the missing elements is determined by interpolating, according to parameter `NSTEP`, the global nodal point numbers associated to the elements in the generation set.

elem/cntv




```
CNTV
    1     8     0     3     4
    1     5    25    21     2     6    26    22
    4     8     0     3    20
   13    17    37    33    14    18    38    34
   12     8     1     3     1
   53    57    77    73    54    58    78    74
   36     8     0     0     0
   55    59    79    75    56    60    80    76
```

Figure II.7 Solid element generation within a volume (**r**, **s**, **t** tangent to the element local coordinate axes; since the similarity option is active in direction 2 (`ISAME` equal to 1 for element 12), elements 1 to 4, 13 to 16, and 25 to 28 are considered as master elements, i. e. the structural matrices associated to elements 29 and 33 for instance are assumed to be identical to the local matrices related to element 25).

II.4.7.4 *Rotating shaft element connectivities* (5I5 / 4I5)

Skip this section if no shaft element set is defined on dataset `elem/ELST`.

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
NODE1 NODE2 NODE3 NODE4
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[2,4]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,1]	(4)	Generation option 0 no generation 1 1-D block generation (straight line)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`NODE1`	1–5	[1,`NNODE`]	(5)	Global node number of element nodal point 1
…				
`NODE4`	16–20	[1,`NNODE`]	(5)	Global node number of element nodal point 4

(1) First element of set must be element number 1; last element, corresponding to element number `Nelem` of current set (see section `elem/ELST`), must be input and may not be generated. Other elements may be defined in any order sequence.

(2) The number of nodes used to describe rotating shaft elements (cylindrical and tapered shapes) is shown in Table II.3. By default, the values given in that table are used. See also Figure II.4g.

elem/cntv

(3) If current element is considered as a slave one, the local stiffness, mass and gyroscopic matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since structural matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) Two elements must be defined for line generation (Fig. II.8). The number of generated elements, as well as their numbering, is determined according to the global nodal point increment used to compute node numbers of missing elements. Note that parameter `NSTEP` must be compatible with the generation scheme and that the nodes may be generated increasingly (`NSTEP` positive) or decreasingly (`NSTEP` negative). Second element in the generation set may be used as first element of a new generation set. If the similarity option is active, the stiffness, mass and gyroscopic matrices of the generated elements are assumed to be identical to those of the first element of the generation set.

(5) The input sequence that must be followed by element node input is given in Figure II.4g. Only the first `NNOEL` global node numbers are read. It should be noted that all the nodes of a shaft element must initially be aligned with the rotation axis of the gyroscopic system and that all the local *r*-axes must be directed towards the same direction. Moreover, for a tapered shaft element, all the nodes must be located on the segment connecting local node `NODE1` of first element in current set to local node `NODE2` of last element. If the generation option is used, the node numbering of the missing elements is determined by interpolating, with reference to parameter `NSTEP`, the global nodal point numbers of the two elements in the generation set.

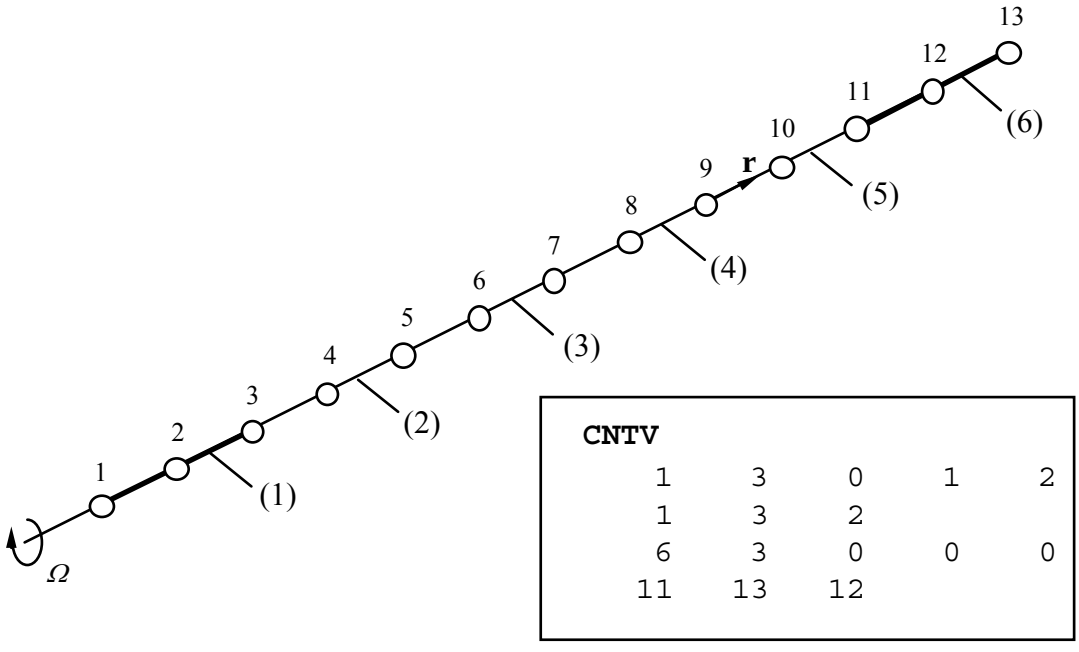


Figure II.8 Rotating shaft element generation along a straight

line (**r** parallel to the centroidal axis and the rotation axis).

II.4.7.5 *Solid to shell transition element connectivities* (5I5 / 15I5)

Skip this section if no solid to shell transition element set is defined on dataset `elem/ELST`.

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
NODE1 NODE2 NODE3 NODE4 NODE5 NODE6 ...      NOD15
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[5,15]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,1]	(4)	Generation option 0 no generation 1 1-D block generation (line)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`NODE1`	1–5	[1,`NNODE`]	(5)	Global node number of element nodal point 1
…				
`NOD15`	71–75	[1,`NNODE`]	(5)	Global node number of element nodal point 15

(1) First element of set must be element number 1; last element, corresponding to element number `Nelem` of current set (see section `elem/ELST`), must be input and may not be generated. Other elements may be defined in any order sequence.

elem/cntv

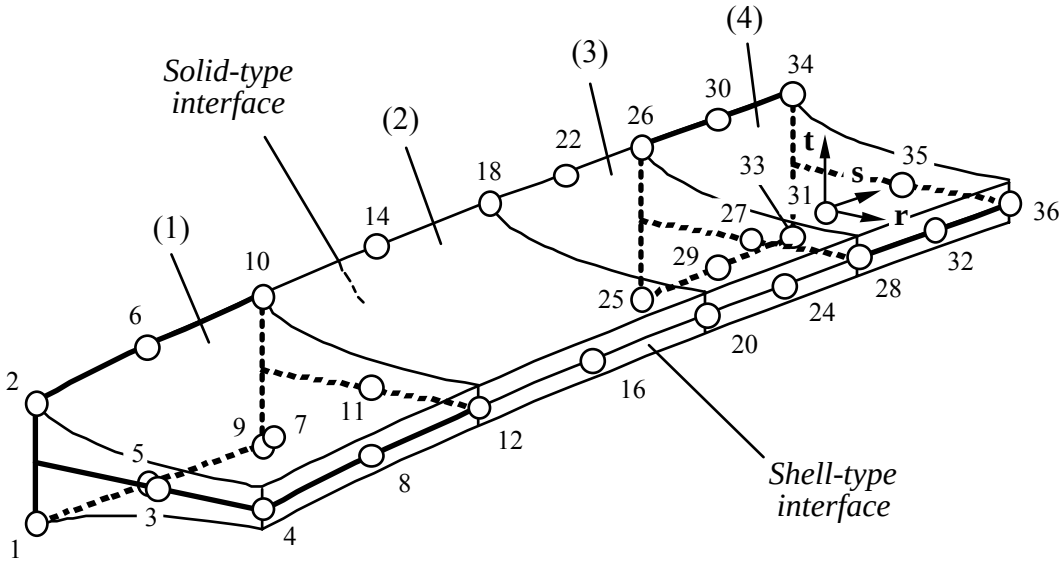
(2) The number of nodes used to describe solid to shell transition elements is shown in Table II.3. By default, the values given in that table are used. See also Figures II.4h and II.4i.

(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since structural matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) Two elements must be defined for line generation (Fig. II.9). The number of generated elements, as well as their numbering, is determined according to the global nodal point increment used to compute node numbers of missing elements. Note that parameter `NSTEP` must be compatible with the generation scheme and that the nodes may be generated increasingly (`NSTEP` positive) or decreasingly (`NSTEP` negative). Second element in the generation set may be used as first element of a new generation set. If the similarity option is active, the stiffness and mass matrices of the generated elements are assumed to be identical to those of the first element of the generation set.

(5) The input sequence that must be followed by element node input is given in Figures II.4h and II.4i. Only the first `NNOEL` global node numbers are read. Note that the solid to shell transition elements are always complete and that the solid-type nodal points are the first to be input, followed by the shell-type nodes. If the generation option is used, the node numbering of the missing elements is determined by interpolating, with reference to parameter `NSTEP`, the global nodal point numbers of the two elements in the generation set.

elem/cntv




```
CNTV
     1    12     0     1     8
     1     9    10     2     5     6     4    12     3     8    11     7
     4    12     0     0     0
    25    33    34    26    29    30    28    36    27    32    35    31
```

Figure II.9 Solid to shell transition element generation along a line (**r**, **s** tangent to the element local coordinate axes, **t** normal to the mid-surface).

elem/cntv

II.4.7.6 *Solid to beam transition element connectivities* (5I5 / 12I5)

Skip this section if no solid to beam transition element set is defined on dataset `elem/ELST`.

```
CNTV
Ielem NNOEL ISAME IGENE NSTEP
ANODE NODE1 NODE2 NODE3 NODE4 NODE5 ...      NOD12
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NNOEL`	6–10	[5,12]	(2)	Number of nodes used to describe current element (Default : see note)
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`IGENE`	16–20	[0,1]	(4)	Generation option 0 no generation 1 1-D block generation (line)
`NSTEP`	21–25	[–9999,99999]	(4)	Node number increment (Default : 1)
`ANODE`	1–5	[1,`NNODE`]	(5)	Auxiliary node in the (*r*-*s*) local plane used for the beam-type nodal points
`NODE1`	6–10	[1,`NNODE`]	(6)	Global node number of element nodal point 1
…				
`NOD12`	61–65	[1,`NNODE`]	(6)	Global node number of element nodal point 12

(1) First element of set must be element number 1; last element, corresponding to element number `Nelem` of current set (see section `elem/ELST`), must be input and may not be generated. Other elements may be defined in any order sequence.

(2) The number of nodes used to describe solid to beam transition elements is shown in Table II.3. By default, the values given in that table are used. See also Figures II.4j to II.4l.

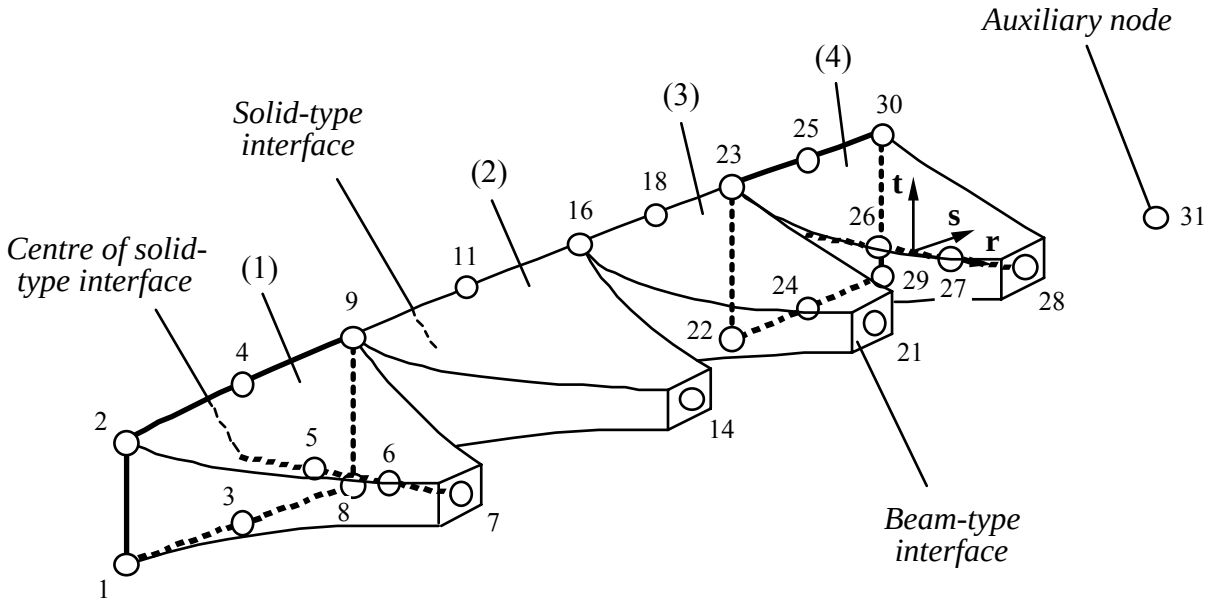
(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease computation time, since structural matrices are only calculated for master elements; it is important to note that the computer code does not check the geometry of elements which are declared by the user as identical. First element of set must always be a master one.

(4) Two elements must be defined for line generation (Fig. II.10). The number of generated elements, as well as their numbering, is determined according to the global nodal point increment used to compute node numbers of missing elements. Note that parameter `NSTEP` must be compatible with the generation scheme and that the nodes may be generated increasingly (`NSTEP` positive) or decreasingly (`NSTEP` negative). Second element in the generation set may be used as first element of a new generation set. If the similarity option is active, the stiffness and mass matrices of the generated elements are assumed to be identical to those of the first element of the generation set.

(5) The auxiliary node is merely used to define the (*r*-*s*) plane used for the beam-type nodal points of the element, i.e. the directions of the principal axes which are necessary to define the cross-sectional dimensions of the element (see section `elem/DIME`). Note that any appropriate node that is defined for the elements can be chosen as auxiliary nodal point. If an additional node out of the element assemblage is defined, all degrees of freedom associated to that node must be deleted using the nodal boundary condition code `ICODE` on dataset `NODE/BCND`. If the line generator is active, the auxiliary nodes associated to the generated elements are interpolated from the auxiliary nodal points of the two elements defined in the generation set; however, the same auxiliary node can be used for all the generated elements. It is important to note that the *s*- and *t*-directions chosen must correspond to the local *s*- and *t*- directions at the solid-type interface of the element; see Figures II.4j to II.4l and Figure II.10 for the construction of the local axes.

(6) The input sequence that must be followed by element node input is given in Figures II.4j to II.4l. Only the first `NNOEL` global node numbers are read. Note that the solid to beam transition elements are always complete and that the solid-type nodal points are the first to be input, followed by the beam-type nodes. Moreover, it should be pointed out that the beam-type nodes, the centre of the solid-type interface and the auxiliary nodal point must initially lie in the (*r*-*s*) plane of the element. If the generation option is used, the node numbering of the missing elements is determined by interpolating, with reference to parameter `NSTEP`, the global nodal point numbers of the two elements in the generation set.

elem/cntv




```
CNTV
     1     9     0     1     7
    31     1     8     9     2     3     4     7     5     6
     4     9     0     0     0
    31    22    29    30    23    24    25    28    26    27
```

Figure II.10 Solid to beam transition element generation along a line (**r** tangent to the pseudo-centroidal axis, **t** cross-product of **r** and the vector from nodal point `NODE1` to the auxiliary node `ANODE`, **s** cross-product of **t** and **r**).

II.4.7.7 *User-defined element connectivities* (4I5 / 6I5)

Skip this section if no user-defined element set is defined on dataset `elem/ELST`.

```
CNTV
Ielem NMUSR ISAME MTAPE
IMUSR INODE IDIRC IGENE MSTEP NSTEP
```

Variable	Columns	Value	Note	Description
`Ielem`	1–5	[1,`Nelem`]	(1)	Element number
`NMUSR`	6–10	[1,99999]	(2)	Number of master degrees of freedom of current user-defined element
`ISAME`	11–15	[0,1]	(3)	Similarity code 0 master element 1 slave element
`MTAPE`	16–20	[0,99]	(4)	Data file unit
`IMUSR`	1–5	[1,`NMUSR`]	(5)	Master degree of freedom number
`INODE`	6–10	[1,`NNODE`]	(6)	Node number
`IDIRC`	11–15	[1,6]	(6)	Global degree-of-freedom number associated to current master degree of freedom
`IGENE`	16–20	[0,2]	(7)	Generation option 0 no generation 1–2 generation
`MSTEP`	21–25	[–9999,99999]	(7)	Master degree-of-freedom number increment (Default : 1)
`NSTEP`	26–30	[–9999,99999]	(7)	Node number increment (Default : 1)

(1) User-defined elements must be input in ascending order and may not be

generated.

elem/cntv

(2) Variable NMUSR corresponds to the order of the stiffness and mass matrices associated to the current user-defined element. Note that gyroscopic or dynamic correction matrices can not be directly specified by the user.

(3) If current element is considered as a slave one, the local stiffness and mass matrices of the element are assumed to be identical to those of the previously defined element, which may be in its turn a slave or master one. The similarity code is introduced to decrease storage requirements, since structural matrices must be stored only for master elements. Note that first element of set must always be a master one. Obviously, a master user-defined element and its associated slave elements must possess the same number of master degrees of freedom.

(4) Parameter MTAPE defines the file unit storing the components of the stiffness and mass matrices of the current master user-supplied element. Only the upper triangular structural matrices must be stored line by line (the first component of each line being the diagonal term) with the format (4 (1X, F19.0)). If parameter IMASS on dataset PROB/MASS is less than 3 indicating that the mass matrix is lumped, only NMUSR components must be given (same format as previously).

The input must be provided in the ascending order of the nodal points and degree-of-freedom directions, since the master degrees of freedom are rearranged in that sequence by the code. Note that each element information, i. e. the set of the stiffness matrix components followed by the mass matrix coefficients respectively, must be preceded by a record containing the keyword elem (format (1X, A4); furthermore, each set of structural data must be preceded in its turn by a record containing the keyword STIF or MASS (same format as previously) respectively. Both structural matrices must not be stored, the omission of the keyword indicating that the corresponding matrix is not provided by the user.

It should be noted that the file units 50 (input), 60 (output), 70 (orthogonalization of the Ritz vectors in the eigensolution schemes) and 71 (starting iteration vectors for the eigensolution schemes) are reserved by the code. Furthermore, file units ITAPE and JTAPE may not be used (see section PROB/PRNT and PROB/PLOT). Note also that variable MTAPE associated with slave elements (see note 3) is ignored by the code.

(5) The number of master degrees of freedom to be defined (directly or by means of the generation option) must be equal to NMUSR. Last degree of freedom that is input must be NMUSR, and no more sets are read after this entry. If in the input sequence the information for a degree of freedom is repeated, the last read or generated information defines the final values associated to that degree of freedom. The sets must not be input in degree-of-freedom order sequence.

(6) Degrees of freedom deleted with the master degree-of-freedom code MCODE on dataset PROB/MESH may not be used as master ones; however, degrees of freedom fixed by using the nodal boundary condition code ICODE on dataset NODE/BCND are valid, but the corresponding stiffness and mass components, as well as the associated off-diagonal terms of the structural matrices, are ignored by the code without a diagnostic message. For shell elements and shell-type nodes of solid to shell transition elements, the rotational degrees of freedom 4 to 6 are

referenced to the local mid-surface system.

(7) The global degree-of-freedom number is set equal to the direction `IDIRC` defined for the first or second node of the generation set, according as parameter `IGENE` is equal to 1 or 2 respectively. Obviously, variables `MSTEP` and `NSTEP` must be compatible with the generation scheme. Note that the master degrees of freedom and the nodes may be generated increasingly or decreasingly, and that the second master degree of freedom of the generation set may be used as first one for a new generation scheme.

elem/blck

II.4.8 *Element connectivities – special block generator* (7I5 / 4I5)

Skip this section if the special block generator is not used. See also sections II.4.7.2 *and* II.4.7.3 *for shell and solid element connectivity generations.*

```
BLCK
IBTYP NELMR NELMS NELMT ISAMR ISAMS ISAMT
NODE1 NODE2 NODE3 NODE4
```

Variable	Columns	Value	Note	Description
IBTYP	1–5	[2,3]	(1)	Special block generator type 2 2-D block generator 3 3-D block generator
NELMR	6–10	[1,Nelem]	(2)	Number of elements to be generated in the *r*-direction of the generation set
NELMS	11–15	[1,Nelem]	(2)	Number of elements to be generated in the *s*-direction of the generation set
NELMT	16–20	[1,Nelem]	(2)	Number of elements to be generated in the *t*-direction of the generation set
ISAMR	21–25	[0,1]	(3)	Similarity code in the *r*-direction 0 option is not active 1 option is active
ISAMS	26–30	[0,1]	(3)	Similarity code in the *s*-direction 0 option is not active 1 option is active
ISAMT	31–35	[0,1]	(3)	Similarity code in the *t*-direction 0 option is not active 1 option is active

(cont.)

elem/blck

(cont.)

Variable	Columns	Value	Note	Description
NODE1	1–5	[1,NNODE]	(4)	Global node number of first corner of generated block
NODE2	6–11	[1,NNODE]	(4)	Global node number of second corner of generated block
…				
NODE4	16–20	[1,NNODE]	(4)	Global node number of fourth corner of generated block

(1) The special block generator constitutes a simple but labor-saving scheme for the generation of regular 2-D and 3-D grids (Figs. II.11 and II.12). It can be used for the block generation of lagrangian (220, 230, 240) or serendipian (231, 241) quadrangular superparametric shell elements (2-D block generator) and lagrangian (320, 330) or serendipian (331) brick solid elements (3-D block generator).

With serendipian elements, the internal nodes of the corresponding lagrangian elements must be present in the mesh in order to obtain a proper element generation, but the computer code deletes automatically the degrees of freedom of these additional nodes.

(2) The total number of elements to be generated, i. e. the product of the variables NELMR, NELMS and possibly NELMT, must be equal to Nelem as given on dataset elem/ELST. When the 2-D block generator (shell elements) is active, parameter NELMT is not used. Note that in order to reduce the generation level (1-D block generation of shell elements and 1-D or 2-D block generation of solid elements), the number of elements to be created in one or more directions may be set to 1.

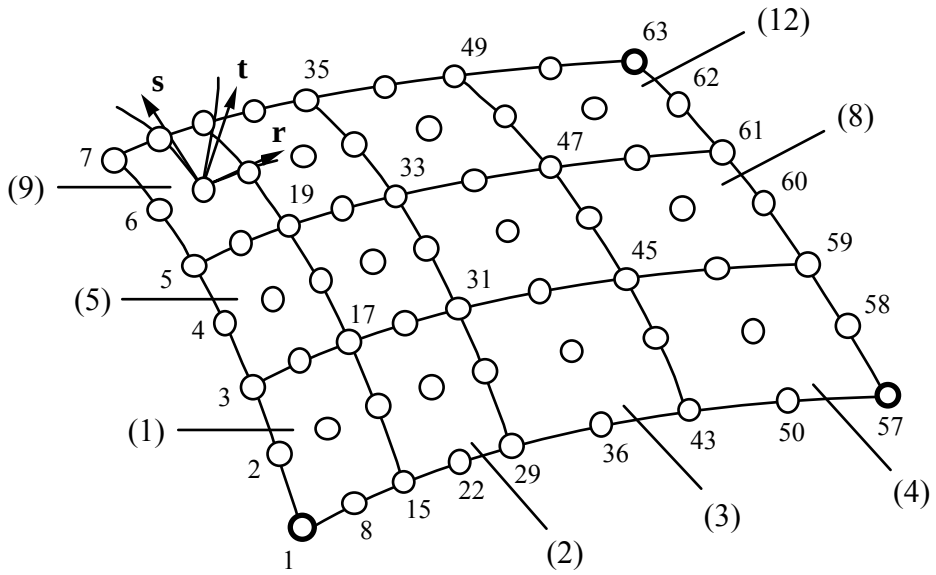
(3) If the similarity option is active in a given direction, the local stiffness and mass matrices of the elements generated in that direction are assumed to be identical to those of the front element, which is then considered as a master element (see Fig. II.12). Note that the similarity code is introduced to decrease computation time, since local stiffness and mass matrices are only calculated for master elements. Obviously, if the similarity option is not active in a given generation direction, all the elements generated in that direction are considered as master elements. When the 2-D block generator (shell elements) is active, parameter ISAMT is not read.

(4) Only the global numbers of three (2-D block generator) or four (3-D block generator) corner nodes of the surface or volume to be created must be given (see Figures II.11 and II.12). Obviously, the numbering of these nodal points must be compatible with the number of elements to be generated in each direction. Note

that the numbering sequence of the corner nodes defines the orientation of the local *r*-, *s*- and *t*-axes, i. e. the *r*-, *s*- and *t*-coordinates are determined from the curved lines joining `NODE1` to `NODE2`, `NODE2` to `NODE3` and `NODE3` to `NODE4` respectively.

elem/blck

The numbers of the elements are allocated successively in the *r*-, *s*- and *t*-directions; consequently, their numbering sequence will not necessarily coincide with the sequence of the node numbers (see Figure II.11). Nevertheless, it is important that the node numbering generates a right-handed local coordinate system.




```
BLCK
     2     4     3     0     0     0     0
     1    57    63
```

Figure II.11 Special 2-D block generation of shell elements (**r**, **s** tangent to the element local coordinate axes, **t** normal to the mid-surface). See also Figure II.6.

elem/blck

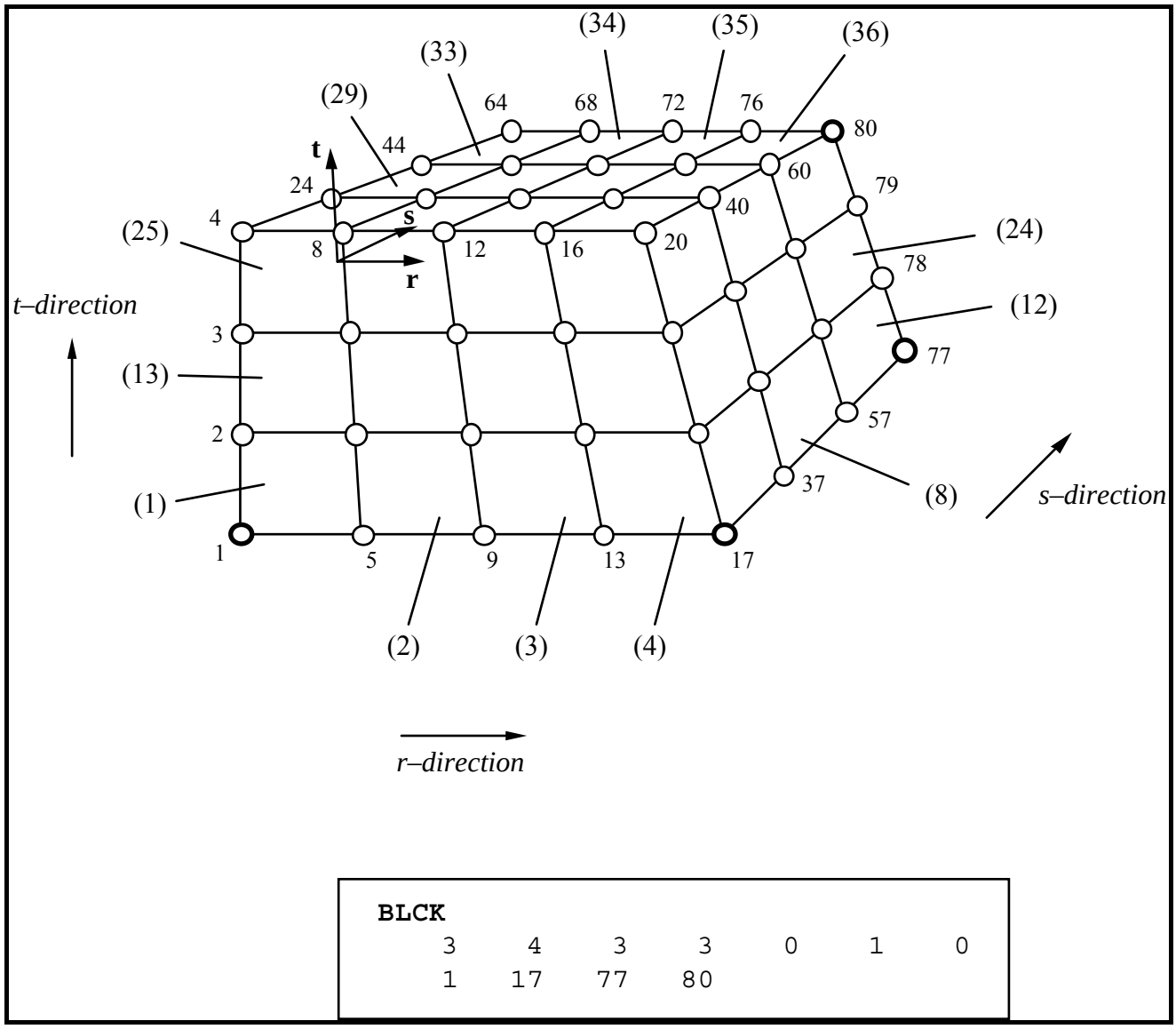


Figure II.12 Special 3-D block generation of solid elements (**r**, **s**, **t** tangent to the element local coordinate axes; since the similarity option is active in the *s*-direction (`ISAMS` equal to 1), elements 1 to 4, 13 to 16, and 25 to 28 are considered as master elements, i. e. the structural matrices associated to elements 29 and 33 for instance are assumed to be identical to the local matrices related to element 25). See also Figure II.7.

vect/subm

II.4.9 *Vectors definition section (Only for synthesis processing)*

```
vect
```

II.4.8 *Number of substructures retained modes* (2I5)

```
subm
isubc nmods
```

Variable	Columns	Value	Note	Description
isubc	1–5	[1,nsubs]	(1)	Substructure number
nmods	6–10	[1,nvect]	(2)	Number of modes retained

(1)

II.4.8 *Selection of substructures retained modes* (3I5)

```
mode
imode igene nstep
```

Variable	Columns	Value	Note	Description
imode	1–5	[1,nmidv]	(1)	Mode number
igene	6–10	[0,1]	(4)	Generation option 0 no generation 1 line generation
nstep	11–15	[–9999,99999]	(5)	Mode number increment for retained mode generation (Default : 1)

(1)

elem/exit

II.4.9 *End of element description section*

```
exit
```

Variable	Columns	Value	Note	Description

II.5 Algorithm prescription section : keyword ALGO

```
ALGO
```

Information block regarding the algorithm prescription must start with the keyword ALGO *and ends with a* STOP *or* EXIT *command (see section II.5.4).*

II.5.1 *Condensation* (*optional* / 2I5, F10.0, 2I5, F10.0 / 6I5)

Skip this section if ICOND *is equal to zero on dataset* PROB/DYNA.

```
COND
NMAST MAXCD CVTOL IERRN IZERO RBMSH
IMAST INODE IDIRC IGENE MSTEP NSTEP
```

Variable	Columns	Value	Note	Description
NMAST	1–5		(1)	Number of master degrees of freedom
		[–9999,–1]		Automatic selection of master degrees of freedom
		[1,99999]		Manuel selection of master degrees of freedom
MAXCD	6–10	[0,99999]	(2)	Maximum number of sub-space iterations allowed for the eigenpair extraction from the quadratic reduced eigenproblem (Default : 24)
CVTOL	11–20	[0.,∞[	(3)	Convergence tolerance for the eigenpair extraction (Default : 1.0E–6)

(cont.)

ALGO/COND

(cont.)

Variable	Columns	Value	Note	Description
IERRN	21–25	[0,1]	(4)	Error norm computation option 0 error norms are not computed 1 error norms are calculated
IZERO	26–30	[0,1]	(5)	Rigid body mode flag 0 no rigid body mode is present 1 rigid body modes are present
RBMSH	31–40	[–∞,0.]	(5)	Rigid body mode shift (Default : see note, unit : (rad/unit-time)^2)
IMAST	1–5	[1,NMAST]	(6)	Master degree-of-freedom number
INODE	6–10	[1,NNODE]	(7)	Node number
IDIRC	11–15	[1,6]	(7)	Global degree-of-freedom number associated to current master degree of freedom
IGENE	16–20	[0,2]	(8)	Generation option 0 no generation 1–2 generation
MSTEP	21–25	[0,99999]	(8)	Master degree-of-freedom number increment (Default : 1)
NSTEP	26–30	[0,99999]	(8)	Node number increment (Default : 1)

(1) The number of master degrees of freedom may not exceed the total number of equilibrium equations of the discrete model. If ITAPE is greater than zero on dataset PROB/PRNT, the master stiffness and mass matrices as given by equations (1.17) and (1.18) in section I.1 are saved in ASCII format on the special print-out unit defined by parameter ITAPE; however, if this unit contains previously stored unformatted information (see sections II.2.5 and II.2.6), the master matrices are not saved. Note that for static condensation without eigenpair extraction (ICOND equal to 1 on dataset PROB/DYNA) the information input after variable NMAST is not used.

If the automatic selection option is active (NMAST negative), the code chooses as master degrees of freedom those generalized nodal displacements which are associated to the largest ratios of the stiffness matrix diagonal terms to the mass matrix ones. The automatic selection scheme adopted by the code distributes appropriately the master degrees of freedom throughout the discrete model, in order to avoid their local concentration. Note that the second dataset regarding the user-defined master degrees of freedom must be omitted when automatic selection is performed.

(2) Variable MAXCD denotes the maximum number of subspace iterations allowed for solving the quadratic reduced eigenproblem (ICOND equal to 3 on dataset PROB/DYNA) by a special subspace iteration scheme adapted to symmetric quadratic eigenproblems with full matrices.

(3) In the solution of the reduced eigenproblem by means of the generalized Jacobi method (static condensation) or the special subspace iteration method (quadratic condensation), convergence on the kth computed eigenvalue ${}^{m}\lambda_{,k}$ is estimated through the expression

$$| {}^{j+1}\lambda^{m}_{,k} - {}^{j}\lambda^{m}_{,k} | \; / \; | {}^{j+1}\lambda^{m}_{,k} | \leq \texttt{CVTOL} \qquad (k = 1, 2, \ldots \texttt{NFREQ})$$

where NFREQ represents the number of requested frequencies (see dataset PROB/DYNA), j denotes the iteration counter and m is the reduction order (number of master degrees of freedom). See also section I.1.

(4) If the error norm computation option is active, the accuracy of the eigenpairs (${}^{l}\lambda^{m}_{,k}$, ${}^{l}\varphi^{m}_{,k}$) computed at the last iteration l is measured by the following estimation:

$$\varepsilon_k = \| \mathbf{K}_m \, {}^{l}\varphi^{m}_{,k} - {}^{l}\lambda^{m}_{,k} \, \mathbf{M}_m \, {}^{l}\varphi^{m}_{,k} - ({}^{l}\lambda^{m}_{,k})^2 \, \mathbf{N}_m \, {}^{l}\varphi^{m}_{,k} \|_2 \; / \; \| \mathbf{K}_m \, {}^{l}\varphi^{m}_{,k} \|_2$$

$$(k = 1, 2, \ldots \texttt{NFREQ})$$

where $\mathbf{K}_m$ and $\mathbf{M}_m$ are the static stiffness and mass matrices as given by equations (1.17) and (1.18) in section I.1, $\mathbf{N}_m$ denotes the quadratic mass matrix defined in equation (1.21), and $\| . \|_2$ symbolizes the Euclidian vector norm. For static condensation, matrix $\mathbf{N}_m$ is obviously omitted.

(5) The eigensolution procedures can calculate frequencies of magnitude zero and their corresponding mode shapes. Such frequencies are present in the model if all rigid body modes have not been suppressed, if hourglass modes (spurious or zero energy modes introduced by the selective or reduced integration scheme) are included in the mesh, or if input or modelling errors are present. Note that the rigid body mode flag must be active if frequencies of magnitude zero are present; otherwise the code stops execution (non-positive definiteness of the reduced stiffness matrix). By default, the program selects as rigid body mode shift the lowest ratio, multiplied by -10^{-3}, of the stiffness matrix diagonal terms to the mass matrix ones. For the eigenpair extraction from the quadratic reduced eigenproblem (dynamic condensation), the magnitude of the rigid body mode shift may not be larger than the absolute value of the first negative eigenvalue of

the shifted condensed stiffness matrix $\mathbf{K}^{\rho}{}_{,m} = \mathbf{K}_m - \rho\,\mathbf{M}_m - \rho^2\,\mathbf{N}_m$, where ρ denotes the rigid body mode shift `RBMSH`; otherwise, matrix $\mathbf{K}^{\rho}{}_{,m}$ becomes again non-positive definite.

ALGO/COND

(6) The number of master degrees of freedom to be defined (directly or by means of the generation option) must be equal to NMAST. Recall that the input is skipped if automatic selection of master degrees of freedom is chosen. Last degree of freedom that is input must be NMAST, and no more sets are read after this entry. If in the input sequence the information for a degree of freedom is repeated, the last read or generated information defines the final values associated to that degree of freedom. The sets must not be input in degree-of-freedom order sequence.

If rigid body modes or hourglass modes (spurious or zero energy modes introduced by the selective or reduced integration scheme) are present, the slave stiffness matrix $\mathbf{K}_{ss}$ as given by equation (1.15) may be singular. This numerical instability can be avoided by selecting at least one master degree of freedom in the direction of each rigid body mode or by choosing as master degree of freedom a non-zero hourglass degree of freedom.

(7) Fixed degrees of freedom (see master degree-of-freedom code MCODE on dataset PROB/MESH and nodal boundary condition code ICODE on dataset NODE/BCND) may not be used as master ones. For shell elements and shell-type nodes of solid to shell transition elements, the rotational degrees of freedom 4 to 6 are referenced to the local mid-surface system.

(8) The global degree-of-freedom number is set equal to the direction IDIRC defined for the first or second node of the generation set, according as parameter IGENE is equal to 1 or 2 respectively. Obviously, variables MSTEP and NSTEP must be compatible with the generation scheme. Note that the master degrees of freedom and the nodes may be generated increasingly or decreasingly, and that the second master degree of freedom of the generation set may be used as first one for a new generation scheme.

II.5.2 ***Eigensolution*** (*optional* / 3I5, F10.0, 3I5, F10.0, I5, 2F10.0, I5)

Skip this section if `IEIGV` *is equal to zero on dataset* `PROB/DYNA`.

```
EIGV
MAXIT NIVEC IIVEC CVTOL IERRN ISSCK IZERO RBMSH  (cont.)
                              (cont.)  IFRTP ENDLT ENDRG
ISVEC
```

Variable	Columns	Value	Note	Description
`MAXIT`	1–5	[0,99999]	(1)	Maximum number of iterations allowed (Default : 24)
`NIVEC`	6–10	[0,99999]	(2)	Number of iteration vectors kept in high speed core (Default : see note)
`IIVEC`	11–15	[0,4]	(3)	Selection option for starting iteration vectors 0 standard selection 1 truncated Lanczos proce- dure 2 pseudo-random vectors 3 random vectors 4 user-defined vectors
`CVTOL`	16–25	[0.,∞[	(4)	Relative convergence tolerance on the eigenvalues (Default : 1.0E–6)
`IERRN`	26–30	[0,1]	(5)	Error norm computation option 0 error norms are not computed 1 error norms are calculated

(cont.)

ALGO/EIGV

(cont.)

Variable	Columns	Value	Note	Description
ISSCK	31–35	[0,1]	(6)	Sturm sequence check option 0 Sturm sequence check is not performed 1 Sturm sequence check is performed
IZERO	36–40	[0,1]	(7)	Rigid body mode flag 0 no rigid body mode is present in the mesh 1 rigid body modes are present in the mesh
RBMSH	41–50	[–∞,0.]	(7)	Rigid body mode shift (Default : see note, unit : $(\text{rad/unit-time})^2$)
IFRTP	51–55	[0,1]	(8)	Frequency mode flag 0 lowest frequencies and associated mode shapes are calculated 1 frequencies and mode shapes are calculated within a prescribed interval
ENDLT	56–65	[0.,∞[	(8)	Left end of prescribed frequency interval (Unit : Hz or unit-time^{-1})
ENDRG	66–75	[0.,∞[	(8)	Right end of prescribed frequency interval (Default : 1.0E+6, unit : Hz or unit-time^{-1})
ISVEC	76–80	[0,1]	(9)	Mode shape storage flag for restart 0 mode shapes are not saved on special print-out unit ITAPE 1 mode shapes are saved on special print-out unit ITAPE

(1) For the standard subspace iteration method (IEIGV equal to 1 on dataset PROB/DYNA), parameter MAXIT denotes the maximum number of subspace iterations allowed up to eigenvalue convergence. With the subspace iteration method accelerated by shifting strategy and overrelaxation (IEIGV equal to 2 on dataset PROB/ DYNA), variable MAXIT limits the number of iterations per shifting block, i. e. the number of iterations between two successive stiffness matrix factorizations. For the Lanczos procedure with selective orthogonalization (IEIGV equal to 3 on dataset PROB/DYNA), parameter MAXIT is related to the Lanczos loops and denotes the dimension of the undeflated tridiagonal matrix. For the quadratic subspace iteration schemes (IEIGV equal to 4 or 5 on dataset PROB/DYNA), variable MAXIT represents the maximum number of subspace inverse or forward iterations allowed up to convergence of the eigenvalues associated with the quadratic eigenproblem arising in the finite dynamic element formulation. For the special subspace iteration method developed for undamped gyroscopic systems (IEIGV equal to 6 on dataset PROB/DYNA), parameter MAXIT limits the number of subspace iterations for the eigenpair extraction at each specified angular speed of the spinning system. For more details, see section I.1.

(2) For the standard, accelerated and special quadratic iteration algorithms, parameter NIVEC is the number of so-called Ritz vectors to be employed simultaneously in the solution scheme. This variable can be smaller (except for the quadratic subspace iteration methods) or larger than the number of required eigenvalues NFREQ defined on dataset PROB/DYNA. During the iteration, all NIVEC iteration vectors are kept in high speed core; hence, a smaller number of subspace iteration vectors will result in less high speed core requirements. By default, NIVEC is set to *min* (2*NFREQ, NFREQ+8). With the standard subspace iteration method, the accelerated scheme is automatically adopted if parameter NIVEC is smaller than this default value. Note that with the accelerated subspace iteration method, the simultaneous iteration is performed only on the subset of iteration vectors which have not fully converged; moreover, during the process, some of these converged iteration vectors may be moved to back-up storage (file unit 70).

For the special subspace iteration method developed for undamped gyroscopic systems, variable NIVEC defines the number of Ritz vectors to be used for the extraction of the eigenpairs of the system at rest (zero spin rate), these eigenpairs being always computed (see dataset ELEM/GYRO). For non-zero angular speeds, parameter NIVEC is implicitly doubled by the code, since the eigenvalues appear in pairs of pure imaginary values (see section I.1).

With the Lanczos procedure, variable NIVEC denotes the number of so-called Lanczos or Krylov vectors kept in high speed core for their selective orthogonalization against the converged eigenvectors. By default, parameter NIVEC is set to MAXIT. If the storage requirements exceed blank common dimension, the Krylov vectors are temporarily moved to back-up storage (file unit 70).

(3) The selection option for the starting iteration vectors is only used in conjunction with the subspace iteration method. Generally, option 0 leads to the best convergence rate. Option 1 is effective when the number of requested frequencies NFREQ is considerably smaller than the number of iteration vectors NIVEC.

Option 2 is based upon a pseudo-random scheme, i. e. the starting iteration vectors remain the same if the problem is rerun. Option 3 (general random number generator) is recommended if the eigensolution scheme fails with the other selection options.

ALGO/EIGV

If an eigensolution was already performed on a structure identical or similar to the one considered in the current analysis, the effectiveness of the subspace iteration method can be increased by providing the already computed eigenvectors as the starting iteration vectors. With option 4, the NIVEC starting vectors are read from file unit 71 with the format (4 (1X, F19.0)). Note that each provided vector must be preceded by a record containing the keyword EIGV (format (1X, A4)) and that the number of components of the vector must be equal to the number of equations in the discrete model (total number of non-zero degrees of freedom). If file unit 71 is empty or contains an insufficient number of vectors, the remaining starting iteration vectors are assumed to be random vectors. For the quadratic subspace iteration schemes, convergence can be accelerated by adding the eigenvalue approximations (shifted by RBMSH if required) on each heading record (format (1X, A4, 1X, F19.0)).

For the special subspace iteration method developed for undamped gyroscopic systems, variable IIVEC is only used for the eigenpair extraction of the system at rest (zero spin rate); for non-zero angular speeds, the eigenvectors computed at the previous stage of the analysis are used as initial vectors.

(4) Convergence on the approximations to the *k*th natural frequency ω_k is estimated through the expression

$$| (\omega^j_{,k}{}^{+1})^2 - (\omega^j_{,k})^2 | \; / \; (\omega^j_{,k}{}^{+1})^2 \leq \texttt{CVTOL} \quad (k = 1, 2, ...\texttt{NFREQ})$$

where NFREQ denotes the number of required frequencies and *j* represents the iteration counter.

Note that, in order to prevent loss of orthogonality of the iteration vectors against converged eigenvectors removed in back-up storage (see note 2 in this section), convergence tolerance is set to *min* (CVTOL, 1.0E–10) if the number of requested frequencies is larger than NIVEC–1 or if the eigenpairs are to be computed within a frequency spectrum of interest (see note 8 in this section). Furthermore, for the procedures based on the subspace iteration method, parameter CVTOL should not be less than 1.0E–12, since the convergence tolerance for the eigenpair extraction in the iterated subspace (generalized Jacobi method) is set to that value.

Parameter CVTOL is always set to the computer round-off error (machine precision) when using the Lanczos procedure with selective orthogonalization.

(5) If the error norm computation option is active, the accuracy of the natural frequencies $\omega^l_{,k}$ and associated mode shapes $\varphi^l_{,k}$ computed at the last iteration *l* is measured by the following estimation:

$$\varepsilon_k = \| \mathbf{K} \, \varphi^l_{,k} - (\omega^l_{,k})^2 \, \mathbf{M} \, \varphi^l_{,k} \|_2 \; / \; \| \mathbf{K} \, \varphi^l_{,k} \|_2 \; (k = 1, 2, ...\texttt{NFREQ})$$

where **K** and **M** are the stiffness and mass matrices of the finite element assemblage, and $\| \, . \, \|_2$ symbolizes the Euclidian vector norm.

If the mesh includes finite dynamic elements, the *k*th error norm at the last iteration *l* is computed as

$$\varepsilon_k = \| \mathbf{K}\, \varphi^{l}_{,k} - (\omega^{l}_{,k})^2\, \mathbf{M}\, \varphi^{l}_{,k} - (\omega^{l}_{,k})^4\, \mathbf{N}\, \varphi^{l}_{,k} \|_2 \;/\; \| \mathbf{K}\, \varphi^{l}_{,k} \|_2$$

($k = 1, 2, \ldots$NFREQ)

where **N** denotes the additional correction matrix of the finite dynamic element assemblage.

For gyroscopic systems, the error estimate for the *k*th mode shape is obtained by the following expression

$$\varepsilon_k = (\textbf{Erreur !})^{1/2}$$

($k = 1, 2, \ldots$NFREQ)

where **G** is the skew-symmetric gyroscopic matrix of the finite element assemblage and ${}^{r}\varphi^{l}_{,k}$ and ${}^{i}\varphi^{l}_{,k}$ represent the real and imaginary parts of the *k*th complex mode shape $\varphi^{l}_{,k}$ computed at the last iteration *l*.

It is strongly recommended to use the error norm option in order to verify the accuracy of the computed eigensolutions and to check if new eigenvalues have suddenly appeared in the converged spectrum during the final control iteration. Finally, it should be mentioned that the computed mode shapes are normalized with respect to the mass matrix **M** or the expanded mass matrix **B** (finite dynamic element formulation and undamped gyroscopic systems) as given by equations (1.23c) and (1.26c) in section I.1.

(6) The Sturm sequence check can be used to verify that all the requested eigenvalues have been computed, i. e. no eigenvalue has been missed in the frequency spectrum of interest. See the flow-charts of the eigensolution procedures presented in section I.1.

(7) The eigensolution procedures can calculate frequencies of magnitude zero and their corresponding mode shapes. Such frequencies are present in the model if all rigid body modes have not been suppressed, if hourglass modes (spurious or zero energy modes introduced by the selective or reduced integration scheme) are present in the mesh, or if input or modelling errors are present. Note that the rigid body mode flag must be active if frequencies of magnitude zero are present; otherwise the code stops execution (non-positive definiteness of the global stiffness matrix). By default, the program selects as rigid body mode shift the lowest ratio, multiplied by -10^{-3}, of the stiffness matrix diagonal terms to the mass matrix ones.

For the quadratic subspace iteration schemes, the magnitude of the rigid body mode shift may not exceed the absolute value of the first negative or secondary eigenvalue of the quadratic eigenproblem as given by equation (1.25) in section I.1; otherwise, the shifted stiffness matrix becomes again non-positive definite

(See section I.1).

ALGO/EIGV

When the special subspace iteration method for undamped gyroscopic system is used, a rigid body mode shift can be applied only if the structure is at rest, i. e. the gyroscopic effects are omitted (zero angular speed). If the singularity of the stiffness matrix is due to the non-gyroscopic effects (axial and torsional vibrations), first compute the natural frequencies of the structure at rest by shifting the eigenvalue spectrum and next calculate the bending modes at a given spin rate by deleting the degrees of freedom corresponding to the axial and torsional vibration modes.

(8) If variable `IFRTP` is equal to zero, the eigensolution algorithm iterates until the smallest `NFREQ` eigenvalues have converged within the selected tolerance `CVTOL`. Note that the code restricts the frequency spectrum to 10^6 Hz or unit-time^{-1}.

If variable `IFRTP` is equal to 1, the code computes the smallest `NFREQ` eigenvalues within an interval which is defined by the lower frequency limit `ENDLT` and the upper frequency limit `ENDRG`. In the calculation of frequencies within an interval, note that the acceleration scheme (shifting strategy and overrelaxation) is always applied and that the maximum number of converged eigenvectors which can be moved to back-up storage (file unit 70) is equal to 20. This option is not allowed if one of the quadratic subspace iteration schemes or the special subspace iteration method for gyroscopic systems is applied.

(9) When the mode shape storage option is active, the eigenvectors are stored on special print-out unit `ITAPE` defined on dataset `PROB/PRNT` (in addition to the listing and plotting files according to the print and plot instructions given on datasets `PROB/PRNT` and `PROB/PLOT`). These vectors may then be retrieved for the starting iteration vectors by setting `IIVEC` equal to 4 in a further analysis. The format used for storage is identical to the one specified in note 3 of this section. Note that if unit `ITAPE` contains previously stored unformatted information (see sections II.2.5 and II.2.6), the mode shapes are not saved.

II.5.3 ***Mode superposition*** (*optional*, *repeatable*)

Skip this section and the next subsections, if `IMSUP` *is equal to zero on dataset* `PROB/DYNA`.

```
MSUP
```

Several mode superposition analyses can be performed in sequence, provided that each information block starts with the keyword `MSUP` *and ends with an* `EXIT` *command (see section II.5.3.9). The eigenpairs (frequencies and mode shapes) extracted by means of the eigensolution algorithms described in the previous section are then assumed to be valid for all the mode superposition cases. Note that if finite dynamic elements are included in the mesh, a mode superposition analysis can not be performed. This is also the case when rotating shaft elements (gyroscopic systems) are present (still under development).*

II.5.3.1 ***Integration scheme*** (I5, 2F10.0, 4I5, F10.0, 4I5, 3X, 2I1)

```
INTE
NMSUP SIGMA ALPHA IDAMP NTIME MAXPT NTSTP DELTA  (cont.)
                      (cont.)  INCND NLOAD IGRND IERRL
IPPDR
```

Variable	Columns	Value	Note	Description
`NMSUP`	1–5	[0,`NFREQ`]	(1)	Number of modes to be used in the mode superposition analysis (Default : 1)
`SIGMA`	6–15	–	(2)	First integration parameter (Default : see note)
`ALPHA`	16–25	–	(2)	Second integration parameter (Default : see note)

(cont.)

MSUP/INTE

(cont.)

Variable	Columns	Value	Note	Description
IDAMP	26–30	[0,2]	(3)	Damping option 0 no damping included 1 Rayleigh damping 2 modal damping
NTIME	31–35	[0,99999]	(4)	Number of time functions
MAXPT	36–40	[0,99999]	(4)	Maximum number of points used to describe any one of the time functions
NTSTP	41–45	[0,99999]	(5)	Number of time steps (Default : 1)
DELTA	46–55	[0.,∞[	(5)	Time step increment (Default : 1.0E–9)
INCND	56–60	[0,3]	(6)	Initial condition flag 0 zero initial displacements and velocities are automatically generated 1 non-zero initial displacements are input 2 non-zero initial velocities are input 3 non-zero initial displacements and velocities are input
NLOAD	61–65	[0,99999]	(7)	Number of sets used to define the concentrated dynamic loads
IGRND	66–70	[0,1]	(7)	Ground acceleration option 0 no ground acceleration is applied 1 mass proportional loading is applied

`IERRL`	71–75	[0,1]	(8)	Error norm option for the external load representation 0 error norms are not calculated 1 error norms are computed at each time step

(cont.)

(cont.)

Variable	Columns	Value	Note	Description
IPPDR	76–80	[0,3]	(9)	Print-out/plotting directive flag (format 2I1) 0 Print-out/plotting (without loading) at each time step and nodal point 1 Print-out/plotting (without loading) at specific time step and/or nodal points 2 Print-out/plotting (loading included) at each time step and nodal point 3 Print-out/plotting (loading included) at specific time steps and/or nodal points

(1) If the number of modes to be used in the mode superposition analysis is larger than the number of converged eigenvalues extracted by the eigensolution algorithm (see section II.5.2), parameter NMSUP is reset to the latter value.

(2) For the Wilson-θ method (IMSUP equal to 1 on dataset PROB/DYNA), variable SIGMA (the classical parameter θ) denotes the integration parameter and should be chosen between 1.37 and 2.01 for an unconditionally stable and accurate integration scheme. By default, SIGMA is set to 1.4. Variable ALPHA is not used when the Wilson-θ method is selected.

For the Newmark integration scheme (IMSUP equal to 2 on dataset PROB/DYNA), variables SIGMA and ALPHA (the classical parameters σ and α) represent the integration parameters and are given for various explicit and implicit schemes by Table I.3 in section I.1. First integration parameter σ should be chosen between 0.5 and 0.55 for an unconditionally stable and accurate scheme. By default, SIGMA is set to 0.5. For the second integration parameter α, it is recommended to select a value between $0.25(\sigma + 0.5)^2$ and $0.5(\sigma + 0.5)$. By default, variable ALPHA is set to the lower limit of the advised values.

(3) If damping is included in the mode superposition analysis, damping information must be input in section ALGO/MSUP/DAMP. With Rayleigh damping, the structural damping matrix **C** as given by equation (1.1) in section I.1 is assumed to be a linear combination of the mass and stiffness matrices of the finite element assemblage. With modal damping, the projection of the structural damping matrix on the modal basis is assumed to be diagonal.

MSUP/INTE

(4) Dynamic loads are applied to the structural system by means of time function tables input in section ALGO/MSUP/TIME and individual load intensity multipliers assigned with the loads (see sections ALGO/MSUP/LOAD and ALGO/MSUP/GRND). Parameter MAXPT denotes the maximum number of pairs of values (time and function value) used to describe any one of the user-defined time functions.

(5) Parameter DELTA denotes the actual constant time step increment used in the step-by-step time integration of the governing equilibrium equations as given by expressions (1.11) to (1.13) in section I.1. This variable, which must be larger than 1.0E–9, defines in conjunction with parameter NTSTP the solution time.

Note that in general the numerical integrations using any of the methods are accurate when the ratio of the time step increment DELTA to the lowest period of the structure included in the mode superposition scheme is smaller than about 0.01. When that ratio is larger, the Newmark constant-average-acceleration method introduces period elongation, whereas the Wilson-θ method introduces amplitude decay in addition to period elongation.

(6) If the non-zero initial condition option is active, the initial nodal displacements and/or velocities corresponding to the time $t = 0.$ are established directly from data input in section ALGO/MSUP/INDS and/or ALGO/MSUP/INVL.

(7) Dynamic loads are defined in section ALGO/MSUP/LOAD and ALGO/MSUP/GRND. Note that variables INCND, NLOAD and IGRND may not be simultaneously equal to zero; otherwise the dynamic response can not be calculated.

(8) The relative error norm for the external load representation in mode superposition analysis is defined, at step i+1, as follows:

$$\varepsilon^{i+1} = \| \mathbf{f}^{i+1} - \textbf{Erreur !}r\textbf{Erreur !}^{+1} (\mathbf{M} \varphi_k) \|_2 \ / \ \| \mathbf{f}^{i+1} \|_2 \qquad (p = \text{NMSUP})$$

where $\mathbf{f}^{i+1}$ denotes the external load at time step i+1, $r^{i}_{,k}{}^{+1}$ represents the kth component of the projection of the applied load on the modal basis at time step i+1, **M** is the mass matrix of the finite element assemblage, φ_k represents the kth mode shape, and $\| \cdot \|_2$ symbolizes the Euclidian vector norm.The error ε^{i+1} is a measure of that part of the external load vector at time t_{i+1} that has not been included in the mode superposition analysis.

(9) First digit of parameter IPPDR defines, according to the print-out codes specified on dataset PROB/PRNT, whether the nodal displacements, velocities and/or accelerations are to be printed entirely or selectively. In the latter case, print-out directives must be defined on dataset ALGO/MSUP/PPDR. If the first digit of IPPDR is larger than 1, the nodal values of the external loading are also printed. With the full print-out option, information at time $t = 0.$ is also printed.

Similarly, second digit of parameter `IPPDR` defines, according to the save codes specified on dataset `PROB/PLOT`, whether the nodal quantities are to be saved on the plot-file at each time step and nodal point or only in a selective way. In the latter case, plotting directives must be specified on dataset `ALGO/MSUP/PPDR`. If the second digit of `IPPDR` is larger than 1, the nodal values of the external loading are also saved on the plot-file (see user's manual of **MAFEPLOT** postprocessor). With the full plotting option, information at time $t = 0.$ is also saved on the plot-file.

MSUP/DAMP

II.5.3.2 ***Damping information*** (*optional* / 8F10.0)

Skip this section if IDAMP *is equal to zero on dataset* MSUP/INTE.

```
DAMP
DAMP1 DAMP2 DAMP3 ...
```

Variable	Columns	Value	Note	Description
DAMP1	1–10	[0.,∞[	(1)	First damping ratio
DAMP2	11–20	[0.,∞[	(1)	Second damping ratio
DAMP3	21–30	[0.,∞[	(1)	Third damping ratio
…				

(1) With Rayleigh damping (IDAMP equal to 1 on dataset MSUP/INTE), the structural damping matrix is assumed to be a linear combination of the mass and stiffness matrices of the finite element assemblage. First and second damping ratios DAMP1 and DAMP2 are then the proportionality constants associated to the global mass and stiffness matrices respectively. Information input on the set after these two variables is not used.

With proportional or modal damping (IDAMP equal to 2 on dataset MSUP/INTE), the modal damping matrix (projection of the structural damping matrix on the modal basis) is assumed to be diagonal. The NMSUP non-dimensional damping ratios ξ_k (k = 1, 2, ...NMSUP) must then be input on one or more sets in ascending order corresponding to the mode numbers (maximum eight damping ratios per record or line input). Note that the kth damping factor (diagonal term of the modal damping matrix) consists of twice the product of the kth damping ratio ξ_k and the natural frequency ω_k of order k. Damping ratios must be input for all NMSUP modes, including those for which damping is neglected. See section I.1.

II.5.3.3 *Time function* (*optional*, *repeatable* / I5, 3F10.0, I5 / 8F10.0)

Skip this section if NTIME *is equal to zero on dataset* MSUP/INTE. *Otherwise, input* NTIME *sets in order of increasing time function number.*

```
TIME
ITTYP TACTV FREQU PHASE NPNTS
TIME1 FVAL1 TIME2 FVAL2 TIME3 FVAL3 ...
```

Variable	Columns	Value	Note	Description
ITTYP	1–5	[1,5]	(1)	Time function type 1 unit step function 2 unit ramp function 3 harmonic function 4 Dirac impulse 5 user-defined function
TACTV	6–15	[0.,∞[	(2)	Time function activity
FREQU	16–20	[0.,∞[	(3)	Frequency of harmonic time function (Unit : Hz or unit-time^{-1})
PHASE	21–30	]–∞,∞[	(3)	Phase angle of harmonic time function (Unit : degrees)
NPNTS	31–40	[0,MAXPT]	(4)	Number of points used to describe current time function (Default : 2)
TIME1	1–10	[0.]	(5)	Time at point 1
FVAL1	11–20	]–∞,∞[	(5)	Function value at point 1
TIME1	21–30	[0.,∞]	(5)	Time at point 2
FVAL1	31–40	]–∞,∞[	(5)	Function value at point 2
TIME1	41–50	[0.,∞]	(5)	Time at point 3
FVAL1	51–60	]–∞,∞[	(5)	Function value at point 3
…				

MSUP/TIME

(1) The information related to each of the NTIME time functions must begin with the keyword TIME**.** Figure II.13 shows the various time function types available in the code.

(2) Parameter TACTV defines the active life of the current time function and, consequently, of the associated dynamic external loads described in sections MSUP/LOAD and MSUP/GRND. That variable is only used for the first three time function types. Note that the end of the time function activity may exceed the solution time and must not necessarily coincide with a multiple of the time step increment DELTA defined on dataset MSUP/INTE.

(3) For harmonic time functions, variables FREQU and PHASE represent the frequency f and phase angle φ, expressed in Hz or (unit-time)$^{-1}$ and degrees respectively, associated to a sinusoidal time function of the form $sin\ 2\pi\ (ft + \varphi/360)$ where t denotes time. These parameters are not read if the time function is not assumed to be harmonic.

(4) Variable NPNTS is only used in conjunction with user-defined time functions and is ignored for other time function types. Note that a minimum of two points is necessary to describe user-defined functions.

(5) Input as many sets in this subsection as are required to define NPNTS points (maximum four points per record or input line). For intrinsically defined time functions (ITTYP less than 5), skip this subsection. Time values at successive points must increase in magnitude. Note that first time value TIME1 must be equal to zero and last time value must be larger than or equal to the end of the solution time. Time values need not be integer multiples of the time step increment DELTA.

MSUP/TIME

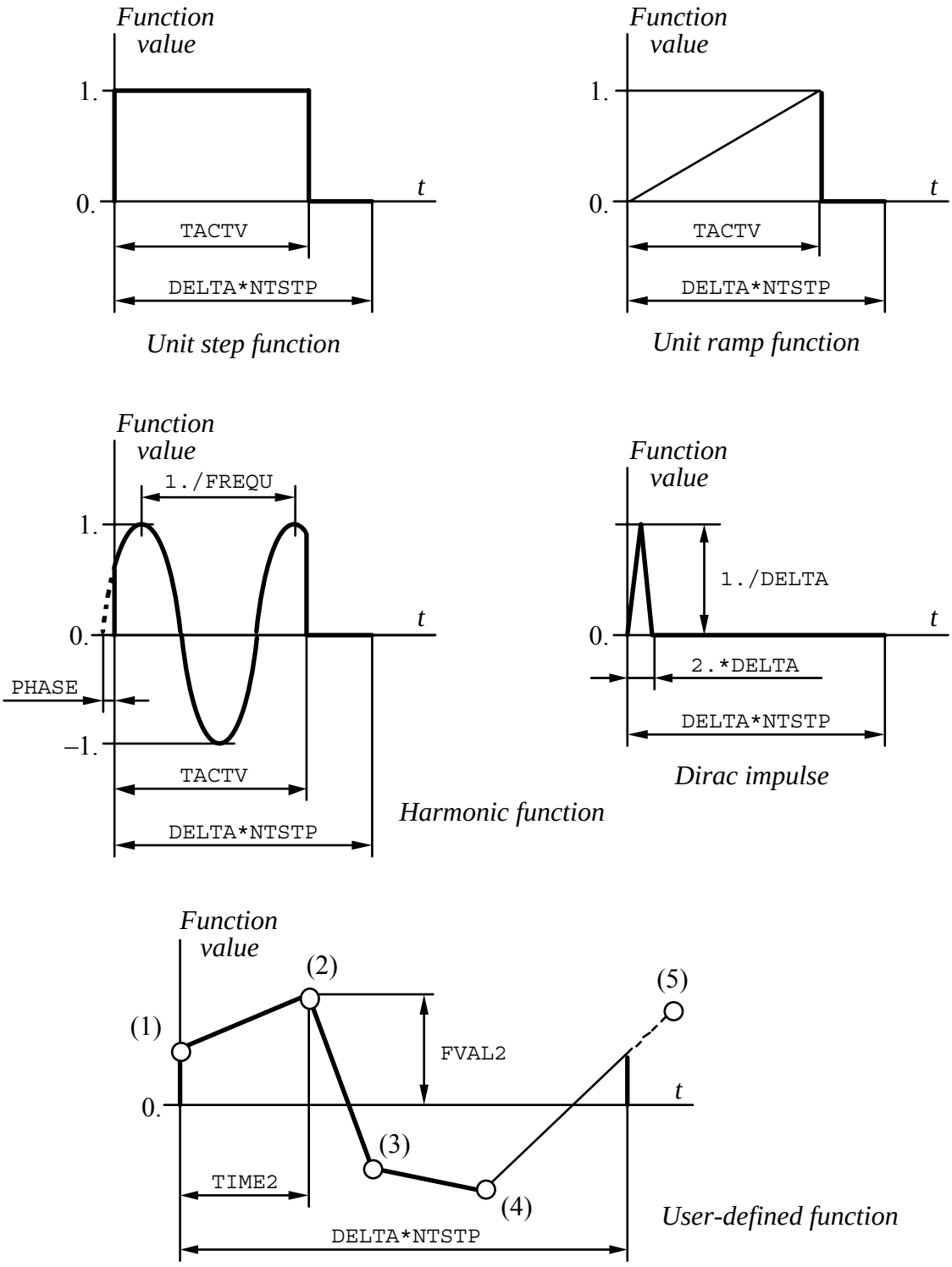


Figure II.13 Time function types available in the code (`DELTA*NTSTP` = solution time).

MSUP/INDS

II.5.3.4 *Initial displacements* (*optional* / I5, 6F10.0, 2I5)

Skip this section if INCND *is equal to zero or 2 on dataset* MSUP/INTE.

```
INDS
INODE XDISP YDISP ZDISP XXDIS YYDIS ZZDIS IGENE NSTEP
```

Variable	Columns	Value	Note	Description
INODE	1–5	[1,NNODE]	(1)	Node number
XDISP	6–15	]–∞,∞[	(2)	*X*-translational displacement
YDISP	16–25	]–∞,∞[	(2)	*Y*-translational displacement
ZDISP	26–35	]–∞,∞[	(2)	*Z*-translational displacement
XXDIS	36–45	]–∞,∞[	(2)	*X*-rotational displacement
XXDIS	46–55	]–∞,∞[	(2)	*Y*-rotational displacement
XXDIS	56–65	]–∞,∞[	(2)	*Z*-rotational displacement
IGENE	66–70	[0,1]	(3)	Generation option 0 no generation 1 generation
NSTEP	71–75	[–9999,99999]	(3)	Node number increment for initial displacement generation

(1) Only non-zero initial displacements need be defined in this section. Nodes that are not mentioned in this section have zero initial displacements assigned. The sets need not be input in node order sequence. The last node that is input must be NNODE. Note that the initial modal accelerations are calculated, with reference to equation (1.10) in section I.1, through the projection on the modal basis of the external loads specified on datasets MSUP/LOAD and MSUP/GRND, the initial modal velocities derived from the information provided on dataset MSUP/INVL and the initial modal displacements obtained from the global translational and rotational displacements given in this section.

(2) The initial displacements input corresponding to deleted degrees of freedom are ignored by the code. If a node is repeated in the input, the initial displacements

specified accumulate at the nodal degrees of freedom. For shell elements and shell-type nodes of solid to shell transition elements, `XXDIS` and `YYDIS` are referenced to the local mid-surface system.

(3) With the generation option, the initial displacements associated with missing nodes are determined by linearly interpolating the information given on the two successive sets used for the generation scheme. Note that the nodes may be generated increasingly or decreasingly and that the node number increment must be compatible with the generation scheme. Second node of the generation set may be used as first node of a new generation scheme.

MSUP/INVL

II.5.3.5 *Initial velocities* (*optional* / I5, 6F10.0, 2I5)

Skip this section if INCND *is equal to zero or 1 on dataset* MSUP/INTE.

```
INVL
INODE XVELO YVELO ZVELO XXVEL YYVEL ZZVEL IGENE NSTEP
```

Variable	Columns	Value	Note	Description
INODE	1–5	[1,NNODE]	(1)	Node number
XVELO	6–15	]–∞,∞[	(1)	*X*-translational velocity
YVELO	16–25	]–∞,∞[	(1)	*Y*-translational velocity
ZVELO	26–35	]–∞,∞[	(1)	*Z*-translational velocity
XXVEL	36–45	]–∞,∞[	(1)	*X*-rotational velocity
XXVEL	46–55	]–∞,∞[	(1)	*Y*-rotational velocity
XXVEL	56–65	]–∞,∞[	(1)	*Z*-rotational velocity
IGENE	66–70	[0,1]	(1)	Generation option 0 no generation 1 generation
NSTEP	71–75	[–9999,99999]	(1)	Node number increment for initial velocity generation

(1) The initial velocity input is analogous to the initial displacement input. See section II.5.3.4 for more details. Note that the initial modal accelerations are calculated, with reference to equation (1.10) in section I.1, through the projection on the modal basis of the external loads specified on dataset MSUP/LOAD and MSUP/GRND, the initial modal displacements derived from the information provided on dataset MSUP/INDS and the initial modal velocities obtained from the global translational and rotational velocities given in this section.

II.5.3.6 ***Dynamic loading*** (*optional* / 3I5, 2F10.0, 2I5)

Skip this section if NLOAD *is equal to zero on dataset* MSUP/INTE.

```
LOAD
INODE IDIRC ITIME AMPLT SHIFT IGENE NSTEP
```

Variable	Columns	Value	Note	Description
INODE	1–5	[1,NNODE]	(1)	Node number to which the load is applied
IDIRC	6–10	[1,6]	(2)	Degree-of-freedom number for the load component
ITIME	11–15	[1,NTIME]	(3)	Time function number that describes the time variation of the load
AMPLT	16–25	]–∞,∞[	(3)	Load amplitude multiplier
SHIFT	26–35	]0.,∞[	(4)	Shift of the time function associated with current load
IGENE	36–40	[0,1]	(5)	Generation option 0 no generation 1 generation
NSTEP	41–45	[–9999,99999]	(5)	Node number increment for load generation

(1) The number of sets (lines) to be input is equal to the value of parameter NLOAD defined on dataset MSUP/INTE. Concentrated load sets need not be input in any node order sequence.

(2) For shell elements or shell-type nodes of solid to shell transition elements, the rotational degrees of freedom 4 to 6 are referenced to the local mid-surface system. Note that if the same degree of freedom of the same node is given a multiple number of times, the code combines the loads algebraically. Loads specified at a deleted degree of freedom are ignored by the program.

MSUP/LOAD

(3) Parameter ITIME indicates the time function employed in specifying the time variation of the concentrated load at time *t*. The load is obtained from the product of the current time function ITIME at time *t* (defined on dataset MSUP/TIME), and the load intensity multiplier AMPLT (Fig. II.14). Note that for a Dirac impulse (ITTYP equal to 4 on dataset MSUP/TIME) the magnitude of the load will be locally equal to the ratio AMPLT/DELTA.

(4) Variable SHIFT denotes the time at which the load starts to be governed by the current time function. Note that the time function may be shifted only forward in time and that the shift must be an integer multiple of the time step increment DELTA.

(5) With the generation option, load data can be generated from the information given on two successive sets, the magnitude of the load associated to missing nodes being derived from the two sets by linear interpolation. Note that variables IDIRC, ITIME and SHIFT for all the generated nodes are taken to be the same as those specified on the first set. Second node may be used as first node of a new generation scheme.

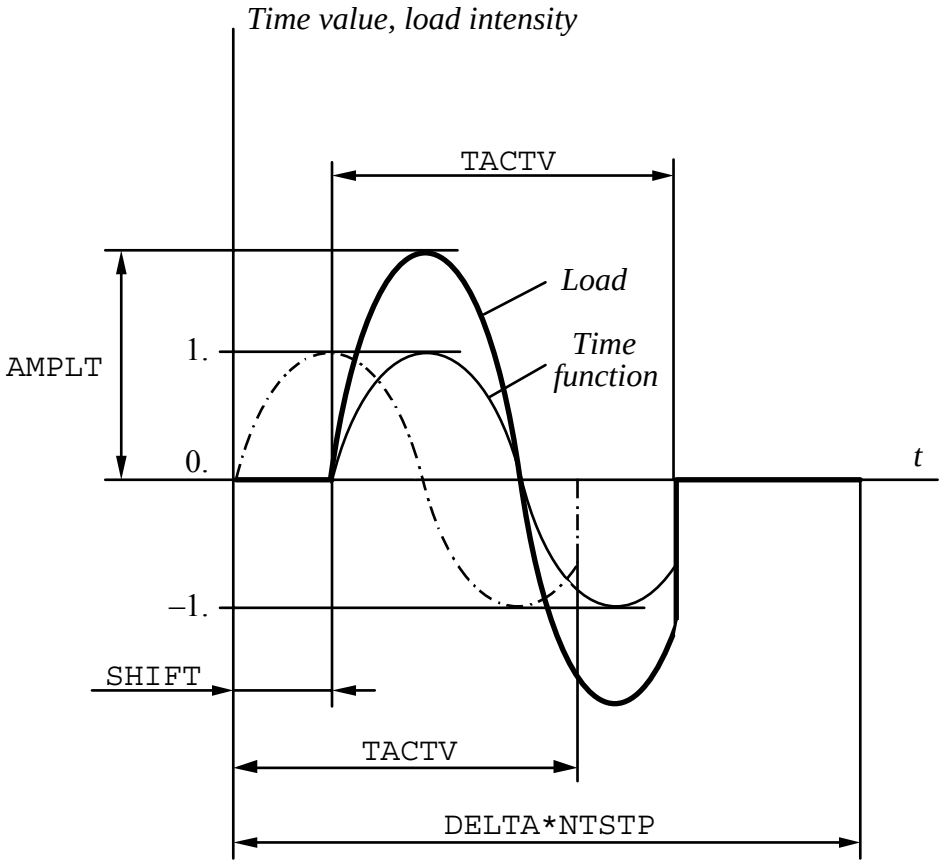


Figure II.14 Concentrated load intensity and shifting of the time function.

II.5.3.7 ***Ground acceleration*** (*optional* / I5, 4F10.0)

Skip this section if IGRND *is equal to zero on dataset* MSUP/INTE.

```
GRND
ITIME XAMPL YAMPL ZAMPL ACCEL
```

Variable	Columns	Value	Note	Description
ITIME	1–5	[1,NTIME]	(1)	Time function number that describes the time variation of the mass proportional loading
XAMPL	6–15	]–∞,∞[	(2)	Load amplitude multiplier in the global *X*-direction
YAMPL	16–25	]–∞,∞[	(2)	Load amplitude multiplier in the global *Y*-direction
ZAMPL	26–35	]–∞,∞[	(2)	Load amplitude multiplier in the global *Z*-direction
ACCEL	36–45	]–∞,∞[	(3)	Acceleration constant

(1) Nodal point loading proportional to the lumped mass or the consistent mass matrix of the finite element assemblage can be used to simulate uniform ground acceleration. Note that only the translational masses are used in the load vector calculation for lumped mass proportional loading (IMASS less than 3 on dataset PROB/ MASS). Parameter ITIME denotes the time function which defines the time variation of the acceleration.

(2) Variable ACCEL is the magnitude of the acceleration constant (expressed in unit of length/(unit-time)^2) and parameters XAMPL, YAMPL and ZAMPL are the load intensity multipliers in the global directions. Note that the sign and the units of ACCEL, as well as the units of the material density DENST assigned to the elements (see dataset ELEM/MATL), must be consistent since the mass proportional loading is determined by their product.

MSUP/PPDR

II.5.3.8 *Print-out/plotting directives* (*optional* / 4I5 / 3I5 / 3I5 / 3I5 / 3I5)

Skip this section if both digits of IPPDR *are equal to zero or* 2 (00, 02, 20 or 22) *on dataset* MSUP/INTE.

```
PPDR
NTIBI NNOBI NTIBJ NNOBJ
ITIMF ITIML ITIMS
INODF INODL INODS
JTIMF JTIML JTIMS
JNODF JNODL JNODS
```

Variable	Columns	Value	Note	Description
NTIBI	1–5	[–1,99999]	(1)	Number of time-step blocks for print-out
NNOBI	6–10	[–1,99999]	(1)	Number of nodal point blocks for print-out
NTIBJ	11–15	[–1,99999]	(1)	Number of time-step blocks for saving on plot-file
NNOBJ	16–20	[–1,99999]	(1)	Number of nodal point blocks for saving on plot-file
ITIMF	1–5	[0,NTSTP]	(2)	First step of current time-step block for print-out
ITIML	6–10	[1,NTSTP]	(2)	Last step of current time-step block for print-out
ITIMS	11–15	[1,99999]	(2)	Step number increment for current time-step block (Default : 1)
INODF	1–5	[0,NNODE]	(2)	First node of current nodal point block for print-out
INODL	6–10	[0,NNODE]	(2)	Last node of current nodal point block for print-out

(cont.)

(cont.)

Variable	Columns	Value	Note	Description
INODS	11–15	[1,99999]	(2)	Node number increment for current nodal point block (Default : 1)
JTIMF	1–5	[0,NTSTP]	(2)	First node of current time-step block for saving on plot-file
JTIML	6–10	[1,NTSTP]	(2)	Last step of current time-step block for saving on plot-file
JTIMS	11–15	[1,999999]	(2)	Step number increment for current time-step block (Default : 1)
JNODF	1–5	[1,NNODE]	(2)	First node of current nodal point block for saving on plot-file
JNODL	6–10	[1,NNODE]	(2)	Last node of current nodal point block for saving on plot-file
JNODS	11–15	[1,999999]	(2)	Node number increment for current nodal point block (Default : 1)

(1) It is usually not necessary to print out and save all the nodal displacements, velocities, accelerations and loads at every solution step. Hence, time steps at which print-out and/or saving on plot-file are desired are grouped into NTIBI and/or NTIBJ blocks respectively. Similarly, nodes at which print-out and/or saving on plot-file are desired are grouped into NNOBI and/or NNOBJ blocks.

If the variable corresponding to the number of time-step and/or nodal point blocks is set to –1, no information is printed and/or saved on plot-file; if that variable is set to zero, print-out and/or saving on plot-file is obtained at each time step and/or nodal point.

Recall that the nodal displacements, velocities and accelerations are printed and saved according to the directives given on datasets PROB/PRNT and PROB/PLOT. Moreover, the external nodal loading is printed or saved with reference to parameter IPPDR specified on dataset MSUP/INTE. Note that the number of time step and nodal point blocks for print-out or saving on plot-file must be equal to zero if the corresponding digit of variable IPPDR is equal to zero or 2, resulting in full print-out or saving on plot-file.

MSUP/PPDR

(2) The NTIBI, NNOBI, NTIBJ and NNOBJ blocks must be successively input in any sequence order of time steps or node numbers. Each of the NTIBI and NTIBJ blocks of time steps is defined by the first and last time steps in the block and the increment to be used to determine the number of time steps between results print-out or saving on plot-file. Similarly, each of the NNOBI and NNOBJ blocks of nodal points is defined by the first and last nodes in the block and the increment to be used to determine the number of nodal results to be printed or saved on plot-file. Note that the information must be skipped if associated number of blocks is less than or equal to zero. Print-out/plotting time steps and node numbers may be generated increasingly or decreasingly.

MSUP/EXIT

II.5.3.9 *End of mode superposition data*

```
EXIT
```

ALGO/STOP

II.5.4 *End of algorithm prescription section*

STOP

Keyword STOP may be replaced by an EXIT command in order to start a new analysis beginning with the keyword PROB.

Variable	Columns	Value	Note	Description

A. APPENDIX 1 : DATA INPUT DIGEST

PROB
TITL
TITLE
MESH
NNODE MCODE NELST NMATL NBCND NMIDV
DYNA
ICOND IEIGV IMSUP NFREQ
MASS
IMASS ICMAS
EXEC
IEXEC IOPTM ISTRT
PRNT
IPRNT ITAPE IPDIS IPVEL IPACC IPMOD
PLOT
JPLOT JTAPE JPDIS JPVEL JPACC JPMOD
EXIT

NODE
COOR
INODE ICTYP XCOOR YCOOR ZCOOR IGENE NSTEP
BCND
INODE ICODE IGENE NSTEP
CSTF
INODE IDIRC STIFF IGENE NSTEP
MIDV
IMIDV IMTYP XMIDV YMIDV ZMIDV THICK IGENE MSTEP
MIDN
INODE IMIDV IGENE NSTEP MSTEP
CMAS
INODE XMASS YMASS ZMASS XINRT YINRT ZINRT IGENE NSTEP
EXIT

ELEM
ELST
IELST IETYP NELEM INTEG IMATL
MATL
IMATL YOUNG POISS DENST SHEAR ISSLW
GEOM
IELST TORSR BENDS BENDT AREAR AREAS AREAT

(cont.)

(cont.)

```
SECT
IELST DMEXL DMINL DMEXR DMINR
GYRO
RPMIN RPMAX RPINC
DIME
IELST DIMS1 DIMT1 DIMS2 DIMT2 DIMS3 DIMT3
CNTV
IELEM NNOEL ISAME IGENE NSTEP
NODE1 NODE2 NODE3 NODE4 NODE5 NODE6 ...
CNTV
IELEM NNOEL ISAME IGENE NSTEP
ANODE NODE1 NODE2 NODE3 NODE4 NODE5 ...
CNTV
IELEM NMUSR ISAME MTAPE
IMUSR INODE IDIRC IGENE MSTEP NSTEP
BLCK
IBTYP NELMR NELMS NELMT ISAMR ISAMS ISAMT
NODE1 NODE2 NODE3 NODE4
EXIT
```

```
ALGO
COND
NMAST MAXCD CVTOL IERRN IZERO RBMSH
IMAST INODE IDIRC IGENE MSTEP NSTEP
EIGV
MAXIT NIVEC IIVEC CVTOL IERRN ISSCK IZERO RBMSH  (cont.)
                                    (cont.)  IFRTP ENDLT ENDRG ISVEC
MSUP
INTE
NMSUP SIGMA ALPHA IDAMP NTIME MAXPT NTSTP DELTA  (cont.)
                                (cont.)  INCND NLOAD IGRND IERRL IPPDR
DAMP
DAMP1 DAMP2 DAMP3 ...
TIME
ITTYP TACTV FREQU PHASE NPNTS
TIME1 FVAL1 TIME2 FVAL2 TIME3 FVAL3 ...
INDS
INODE XDISP YDISP ZDISP XXDIS YYDIS ZZDIS IGENE NSTEP
```

(cont.)

(cont.)

```
INVL
INODE XVELO YVELO ZVELO XXVEL YYVEL ZZVEL IGENE NSTEP
LOAD
INODE IDIRC ITIME AMPLT SHIFT IGENE NSTEP
GRND
ITIME XAMPL YAMPL ZAMPL ACCEL
PPDR
NTIBI NNOBI NTIBJ NNOBJ
ITIMF ITIML ITIMS
INODF INODL INODS
JTIMF JTIML JTIMS
JNODF JNODL JNODS
EXIT
STOP
```